Computer Programming - PASCAL
Enter the following simple Pascal program into your computer:

```
{ My first program }
{ By < YOUR NAME > }
{ on < TODAY’s DATE > }

program my_first_program(input,output);
var
    name:string;
    age:integer;
begin
    writeln(‘Please enter your name’);
    readln(name);
    writeln(‘Please enter your age’);
    readln(age);
    writeln(‘Hello ’,name);
    writeln(‘You are ’,age)
end.
```

Run your program, and write a brief description of what you see on the screen.

The first three lines (in braces) are known as internal documentation, and help others to know what the program does, who wrote it and when it was written.

For any information input, or calculations made, **variables** are used to store this in the computer’s memory.

The main ‘body’ of the program is enclosed between the BEGIN and END statements.

**PROGRAMMING PROBLEM # 1**

Write a program to work out the total number of passengers on an airplane which has double seats and triple seats. The user should enter the number of double seats, and the number of triple seats and the program should calculate the total number of passengers on board the airplane.

**PROGRAMMING PROBLEM # 2**

Write a program to calculate the amount of money earned in a sponsored swim (in dollars) if the person is sponsored at the rate of $0.63 per length of the swimming pool completed.
Type the following program into your computer:

```pascal
{ Program to show use of the IF command }
{ By A. Programmer }
{ on February 2 1999 }

program top_secret(input,output);
var
  password:string;
begin
  writeln('Please enter the password');
  readln(password);
  if password='secret' then
    writeln('Access granted')
  else
    writeln('Danger ! Access denied !')
end.
```

Write a brief description of what this program does.
Type the following program into your computer:

```pascal
program tickets(input, output);
var
ticketnum: integer;
begin
  for ticketnum := 1 to 5 do begin
    writeln;
    writeln('*********************');
    writeln('*   EASTER DISCO    *');
    writeln('*                   *');
    writeln('*    at SPUDZs      *');
    writeln('*      7:30pm       *');
    writeln('* * * * * * * * * * *');
    writeln('* Ticket no. ', ticketnum, '     *');
    writeln('*********************');
    writeln;
  end
end.
```

Write a brief description of what this program does.

Edit the above program so that it will produce ten tickets instead of 5, and will also print out your name as the DJ on each dance ticket.
To simplify programming, **procedures** are used (‘blocks’ of code).

Type the following program into your computer:

```pascal
{ Program to show use of PROCEDURES }  
{ By A. Programmer }  
{ on February 5 1999 }  

program procedures(input,output);  
var  
  hours, rate, wages: real;  

{ This procedure gets the hours worked }  
procedure get_hours;  
begin  
  writeln('Please enter the hours worked');  
  readln(hours);  
end;  

{ This procedure gets the hourly rate }  
procedure get_hourly_rate;  
begin  
  writeln('Please enter the hourly rate');  
  readln(rate);  
end;  

{ This procedure calculates the wages }  
procedure calculate_wages;  
begin  
  wages := hours * rate;  
end;  

{ This procedure displays the wages }  
procedure display_wages;  
begin  
  writeln('Wages earned = $', wages:5:2);  
end;  

{ Main program begins here }  
begin  
  get_hours;  
  get_hourly_rate;  
  calculate_wages;  
  display_wages  
end.
```

Write a brief description of what this program does.
PROGRAMMING PROBLEM # 3

Write a program using procedures which will calculate the length and cost of fencing a farmer needs to buy to go around his field. The unit cost for the fencing is $1 per meter.
It is sometimes necessary to have a program repeat a section of code until certain conditions are met.

Type the program on the following page into your computer, and write a brief description of what this program does.
{ Program to show use of CONDITIONAL LOOPS } 
{ and cumulative totals } 
{ By A. Programmer } 
{ on February 8 1999 }

program conditional_loop(input,output);

var
  data_entries, number, running_total:integer;
  average:real;
  answer:string;

{ This procedure gets the number }
procedure get_number;
begin
  writeln('Please enter a number');
  readln(number);
end;

{ This procedure tracks the number of data entries } 
{ and calculates a running ( cumulative ) total }
procedure track_and_calculate;
begin
  data_entries:=data_entries+1;
  running_total:=running_total+number;
end;

{ This procedure asks the user if there is any more data }
procedure query_user;
begin
  writeln('Is there any more data ?');
  readln(answer);
end;

{ This procedure calculates and displays the average }
procedure calculate_and_display_data;
begin
  average:=running_total/data_entries
  writeln('Total number of entries was ',data_entries);
  writeln('The cumulative total was ',running_total);
  writeln('The average was ',average:5:1);
end;

{ Main program begins here }
begin
  data_entries:=0;
  running_total:=0;
  repeat
    get_number;
    track_and_calculate;
    query_user;
  until answer<>'y';
  calculate_and_display_data
end.
Write a program using procedures which will simulate an electronic scoreboard. The user should be able to choose from a menu either M, H, V or Q.

- if M is pressed, then a message should be displayed, such as “GOAL!” or “Good shot!”
- if H is pressed, then the home team score should be increased by 1
- if V is pressed, then the visiting team score should be increased by 1
- if Q is pressed, then the program should quit
If many conditions have to be tested against, it is better to use a CASE statement than many many many IF..THEN, IF..THEN, IF..THEN statements.

Type the following program into your computer, and write a brief description of what this program does.

```pascal
program month_selector(input,output);
var
    month:integer;

{ This procedure gets the month as a number input }
procedure get_month;
begin
    writeln('Enter a month - between 1 and 12');
    readln(month);
end;

{ This procedure checks the input and displays }
{ the appropriate output }
procedure check_and_display;
begin
    case month of
        1:writeln('That is for January');
        2:writeln('That is for February');
        3:writeln('That is for March');
        4:writeln('That is for April');
        5:writeln('That is for May');
        6:writeln('That is for June');
        7:writeln('That is for July');
        8:writeln('That is for August');
        9:writeln('That is for September');
        10:writeln('That is for October');
        11:writeln('That is for November');
        12:writeln('That is for December');
    otherwise
        writeln('There is no such month !');
    end;
end;

{ Main program begins here }
begin
    get_month;
    check_and_display
end.
```
PROGRAMMING PROBLEM # 5

Write a program using procedures, and the CASE statement which will allow the user to input a student’s grade, and output the appropriate comment which could be printed on the student’s report card.

The comments are based on the following grade structure:

1 : Pass with distinction
2 : Pass with merit
3 : Pass
4 : Marginal fail
5 : Fail
6 : No award
PROGRAMMING PROBLEM # 6

Write a program using procedures which will simulate an electronic cash register as used in 'McDonalds' (feel free to use your own fast food franchise name!).

The person operating the cash register should be able to enter the food items ordered by pressing single keys.

Here is a sample selection of keys and what should happen on pressing each:

- B - burger
- F - fries
- M - shake (strawberry milkshake?)
- S - soda (diet coke?)
- P - pie (apple?)

As each item is entered, a running total should be kept of the amount charged.

When the customer completes his/her order, the cashier should be able to press 'T' to total the amount for the customer to pay.

The cashier should then enter the amount the customer is paying, and the program should calculate and display the change which the customer is to receive (if any).

EXAMPLE SCREEN-SHOT

Hi and welcome to McScott's!
ENTER YOUR ORDER
B <ENTER>
Burger - $3.50
F <ENTER>
Fries - $2.20
S <ENTER>
Diet coke - $1.90
T <ENTER>
YOUR TOTAL IS $7.60
PLEASE ENTER THE AMOUNT TENDERED
10
YOUR CHANGE IS $2.40... HAVE A NICE DAY!
Often, programmers need to store lots of data. To do this would require many variables, but if the data is all of the same format, then ARRAYS can be used.

Type the following program into your computer, and write a brief description of what this program does.

```pascal
{ Program to show use of ARRAYS }
{ By A. Programmer }
{ on February 14 1999 }

program data_storage(input,output);
var
    number:array[1..3] of integer;

{ This procedure gets the numbers input }
procedure get_values;
var
    counter:integer;
begin
    for counter:=1 to 3 do
    begin
        writeln('Enter a value');
        readln(number[counter]);
    end;
end;

{ This procedure outputs the numbers in reverse order }
procedure output_values;
var
    counter:integer;
begin
    for counter:=3 downto 1 do
    writeln(number[counter]);
end;

{ Main program begins here }
begin
    get_values;
    output_values
end.
```

**PROGRAMMING PROBLEM #7**

Write a program **using procedures** which will read in 3 student names entered by the user.

The program should also read in the age of each of these students.

The program should output the name of each student and his / her corresponding age.
PROGRAMMING PROBLEM # 8

Write a program using procedures which will simulate (play) the game of “Scissors, Paper, Stone”. The human should be given the opportunity to pick one of the three options e.g. ‘1’ for scissors, ‘2’ for paper and ‘3’ for stone.

The computer should ‘pick’ one of the three options (see ‘random’ procedure below). The winner is decided according to the following:

- scissors beats paper
- paper beats stone
- stone beats scissors

If the human and the computer both choose the same option, then a draw / tie is declared and a replay is played.

The game should play until one player reaches 10 wins.

The following procedure can be used in your program - it will generate a random number between 1 and 3 (which could be used to represent scissors, paper or stone).

```pascal
procedure pick_random_number;
var
  number:integer;
begin
  getdatetime(randseed);
  repeat
    number:=random;
  until (number>=1) and (number<=3);
  writeln('Random number selected is ',number);
end;
```
Outputting results to the screen (monitor) is useful, but sometimes a hard-copy of program output is also required.

Type the program on the following page into your computer, and write a brief description of what this program does.

PROGRAMMING PROBLEM # 9

Write a program using procedures which will read in the following information about a dance:

- the date
- the time
- the place
- the DJ
- the price

and output a ticket for the dance. You should make some effort to format the output to make it look ‘presentable’ i.e. the ticket should not just have the above information on it - include some asterisks (*), or (#) as ‘borders’.

The program should preview the output to the screen and then offer the user the option of output to the printer to produce a hard copy.
{ Program to show how to produce hard-copy output }
{ By A. Programmer }
{ on February 21 1999 }

program printed_output(input,output);
var
  name:string;
  age:integer;
  choice:char;
  printdevice:text; { THIS LINE IS NEEDED TO PRINT }

{ This procedure gets the input }
procedure get_data;
begin
  writeln('Please enter your name');
  readln(name);
  writeln('Please enter your age');
  readln(age);
end;

{ This procedure lets the user choose screen or printed output }
procedure select;
begin
  writeln('Do you wish output to (s)creen or (p)rinter ?
end;

procedure output_to_screen;
begin
  page(output);
  writeln('Your name is ',name);
  writeln('You are ',age,' years old');
end;

procedure output_to_printer;
begin
  { THIS LINE OPENS UP THE PRINTER DEVICE }
  rewrite(printdevice,'printer:');
  { TO DIRECT OUTPUT TO PRINT, NEED ‘PRINTDEVICE’ AS SHOWN}
  writeln(printdevice, 'Your name is ',name);
  writeln(printdevice, 'You are ',age,' years old.');
end;

{ Main program begins here }
begin
  get_data;
  select;
  if choice = ‘s’ then
    output_to_screen;
  if choice = ‘p’ then
    output_to_printer;
end.
Until now, we have only been working with variables, and operating on data stored in memory of the computer. It is often useful to have a more permanent storage system i.e. saving your work for future use the following day, week, month, year etc.

Type the following program into your computer, and write a brief description of what this program does.

```
{ Program to show how to show file-handling - SAVING }
{ By A. Programmer }
{ on February 28 1999 }

program file_save(input,output);

var
    name:string;
    textfile:text; { THIS LINE IS NEEDED TO SAVE FILES }

{ This procedure gets the input }
procedure get_data;
begin
    writeln('Please enter the name to be saved');
    readln(name);
end;

procedure write_to_file;
begin
    open(textfile,'data');
    writeln(textfile,name);
    close(textfile);
end;

begin
    get_data;
    write_to_file
end.
```

**PROGRAMMING PROBLEM # 10**

Write a program using procedures which will read in three names, and save them in a file called ’names.db’.
Type the following program into your computer.

```pascal
{ Program to show how to show file-handling - LOADING / OPENING }  
{ By A. Programmer }  
{ on March 1 1999 }  

program file_open(input,output);  
var  
    filename:string;  
    textfile:text; { THIS LINE IS NEEDED TO READ FILES }  
    title,author:string;  

{ This procedure gets the name of the file to open }  
procedure load_file;  
begins  
    writeln('Please enter the name of the file you wish to open');  
    readln(filename);  
end;  

{ This procedure reads in data from the file and displays on screen }  
procedure read_from_file;  
var  
    counter:integer;  
begins  
    open(textfile,filename);  
    for counter:=1 to 3 do  
    begin  
        readln(textfile,title);  
        writeln('Book title is ',title);  
        readln(textfile,author);  
        writeln('Author is ',author);  
    end;  
    close(textfile);  
end;  

begin  
    load_file;  
    read_from_file  
end.
```

Create and save a basic text file containing the text shown here:

```
Tales of Crime
Ivor Alibi
15 Minute Dishes
Can U Cooke
Opticians Handbook
I C Clearly
```
using SimpleText - IN ORDER FOR THIS ASSIGNMENT TO WORK, THE TEXT FILE MUST BE IN THE SAME FOLDER WHICH CONTAINS YOUR PROGRAM!
(MAKE SURE YOU PRESS RETURN AFTER THE LAST NAME TO PLACE THE CURSOR BELOW THE TEXT, AFTER THE LAST LINE)

Run the program and write a brief description of what this program does.

PROGRAMMING PROBLEM # 11

Write a program using procedures which will read in a textfile containing data about five flights, and display the data on screen. A sample text file showing two flights is shown here (created in SimpleText):

British Airways
BA635
Larnaca
London
Aeroflot
AF464
Moscow
Larnaca

NOTE: The ‘fields’ (variables) in the text listed above are

AIRLINE
FLIGHTNUMBER
DEPARTURE
DESTINATION
PROGRAMMING PROBLEM # 12

Write a program using procedures which will offer a menu of choices:

1. Create a database file
2. Load an existing database file
3. Print a database file

When the user selects one of the above choices, the appropriate action is taken. For example, if he/she presses ‘1’, then they can create a database on a subject of their choice.

If the user presses ‘2’, they could then load the database from the file they created earlier by pressing ‘1’.

If the user presses ‘3’, then they can print the database file.

You can set up the program to create a database on a topic of your choice - IT MUST CONTAIN AT LEAST 2 FIELDS.
PROGRAMMING PROBLEM # 13

A savings and loan association determines the maximum amount it can loan to an individual for a home mortgage using the following system:

- The individual’s job and the home being purchased are each rated on a scale from 1 to 10
- If the job rating is 1, no loan is given
- If the job rating is greater than 1 but less than 7, the loan limit is determined by the formula:
  \[
  \left(\frac{\text{jobrating}}{10}\right) \times \text{annualsalary} \times 2 \times \left(\frac{\text{homerating}}{10}\right)
  \]
- If the job rating is 7 or more, the mortgage limit is determined by:
  \[
  \left(\frac{\text{jobrating}}{10}\right) \times \text{annualsalary} \times 3 \times \left(\frac{\text{homerating}}{10}\right)
  \]

Write a program using procedures that computes the loan limit based on a customer’s ratings and annual salary.

PROGRAMMING PROBLEM # 14

Write a program using procedures that passes an imaginary token back and forth among four potential owners named A, B, C and D under the restriction that the token can have only one owner at any given time. In particular, a request to give the token to a letter is implemented by typing that letter at the keyboard. If the token is available, your program should respond with the message:

Token is now assigned to X

- where X is the letter that was typed. On the other hand, your program should respond with the message “Token is not available” if the token is not available (if it is currently owned by another letter). Typing the letter currently owning the token should cause the token to be released and the program to respond with the message “Token is now available”.

PROGRAMMING PROBLEM # 15

Write a program using procedures that simulates the operation of an ATM (automated teller machine) - like the electronic cash points offered by Popular Bank, Cyprus Bank etc.

The first step should be the user should input their PIN (Personal Identification Number) - this is a four-digit number known only to the customer. He/she is only allowed 3 possible tries before the ATM shuts down - this is to prevent fraud.

Next, assuming the customer enters his/her PIN correctly, a menu of choices should be made available:

- find out balance (the amount of money in their account)
- withdraw cash (fixed amounts - £10, £20, £40, £60, £80, £100 or other amount but only in multiples of £10 AND the customer should not be able to withdraw more money than is in their account, the balance should be reduced by the appropriate amount)
- make a deposit, either check or cash (cash in £’s only, check for any amount which should be credited to the customer’s account i.e. the balance should be increased by the amount deposited)
- send the bank a message
- end transaction, withdraw card

As an extension to this task, you could have the ATM print out a receipt.
(a) This loop is intended to add the first ten numbers. Will it? Why not?

```pascal
count:=1;
sum:=0;
while count<10 do
  begin
    count:=count+1;
    sum:=sum+count;
  end;
```

(b) The number of fish in Lake Lackluster is currently about ten million. Write a loop that determines how long it will take the population to drop to a tenth of this level if the number declines by 2.3% each year.

(c) **Pick-up-stones** is an easy game of position. Twenty one stones are put in a pile, and players take turns removing 1, 2 or 3 stones from the pile. The last player to go wins. Write a program that allows a human to play this game against the computer. Your program will need to keep track of how many stones are remaining, and also have the computer randomly select 1, 2 or 3 stones when it is the computer’s turn (refer to the previous example of ‘Scissors, paper, stone’).

(d) Adapt your “Guess the number” / “Too high, too low” program so that a number between 1 and 100 is randomly selected by the computer rather than being fixed in the program code.

(e) The following statement appears on a programming final exam:

“Make a statement. If the statement is true, you fail the exam. If the statement is false you fail the quarter.”

What’s the correct answer to avoid failing?
When you press letters on the computer keyboard, the computer does not ‘understand’ the letters you press - it converts the letters to a number based on a special set of codes, like a lookup table.

Type the following program into your computer, and write a brief description of what this program does.

```pascal
{ Program to show use of letter to number conversion } 
{ By A. Programmer } 
{ on March 26 1999 } 

program letter_conversion(input, output);
var
    key, answer;
    number: integer;

{ This procedure gets the key_pressed }
procedure get_key;
begin
    writeln('Please press a key');
    readln(key);
end;

{ This procedure converts the key to its numeric equivalent }
procedure convert_key;
begin
    number := ord(key);
end;

{ This procedure outputs the results }
procedure output_result;
begin
    writeln('The numeric equivalent for ', key, ' is ', number);
end;

{ Main program begins here }
begin
    repeat
        get_key;
        convert_key;
        output_result;
        writeln('Go again ?');
        readln(answer);
        until (answer = 'n') or (answer = 'N')
end.
```
PROGRAMMING PROBLEM #17

By referring to your previous programs on random number generation, write a program (using procedures) which could be used as part of a word game. Your program should generate 7 random letters of the alphabet which players could then be challenged to come up with as many words from as they can in a set amount of time.

Your program will need to generate random numbers between 65 and 90 - and with conversion, will output the corresponding letters.

[NOTE: 65 represents upper case ‘A’, 90 represents upper case ‘Z’]

This section of code (as used in the previous program) will convert a letter - key pressed - to its numeric equivalent:

```pascal
procedure convert_key;
begin
  number:=ord(key);  { 'ord' finds the number for }  
end;  { any key pressed }
```

This section of code will convert a number to its alphabetic equivalent:

```pascal
procedure convert_number;
begin
  letter:=chr(number);  { 'chr' finds the letter or }  
end;  { character associated with }  
{ a number - range from 0 - 255 }