

SULIT



UNIVERSITI TUN HUSSEIN ONN MALAYSIA

**FINAL EXAMINATION
SEMESTER II
SESSION 2011/2012**

SUBJECT NAME : COMPUTER PROGRAMMING
SUBJECT CODE : BTI 10202
COURSE : 1 BDD
EXAMINATION DATE : JUNE 2012
DURATION : 1 HOUR
INSTRUCTION : ANSWER ALL THE 35 QUESTIONS

PLEASE RETURN BACK THIS QUESTION PAPER

THIS EXAMINATION PAPER CONTAIN TEN (10) PAGES

SULIT

- Q1. What is the correct arrangement of six phases for the typical program development?
- Edit/write → Preprocess → Compile → Link → Load → Execute
 - Edit/write → Preprocess → Link → Compile → Load → Execute
 - Edit/write → Preprocess → Link → Load → Compile → Execute
 - Edit/write → Preprocess → Link → Compile → Execute → Load
- Q2. Which statement is not true about *Preprocessing*?
- Write a program to compile the program.
 - Preprocessor program executes manually and process the program code.
 - The Preprocessor obeys the commands from preprocessor directives.
 - Preprocessor occurs before a program is compiled.
- Q3. During compiling a program, if unsuccessful, _____ will be displayed in the compiling dialogue box. The process of correcting this problem is call _____.
- Object code, link
 - logic error, debugging
 - Compile, edit
 - Execute, run
- Q4. All the answers below represent the types of common programming error except;
- Syntax error
 - Logic error
 - Run - time error
 - debugging error
- Q5. Which statement is true about *Logic error*?
- Error occurs due to inappropriate output.
 - Error occurs due to wrong input by user.
 - Error occurred during compilation normally due to syntax problem.
 - Error occurs due to wrong input by user and during compilation.
- Q6. *Standard header file* and list of *functions* stated as below, except;
- stdio.h – printf(), scanf()
 - conio.h – clrsc(), getch()
 - int.h – float(), double()
 - math.h – sqrt(), pow()
- Q7. Which combination of main function is wrong?
- void main ()

```
{
    return 0;
}
```
 - int main ()

```
{
```

```

return 0;
}

```

c. main ()

```

{
Return 0;
}

```

d. void main ()

```

{..... }

```

- Q8 *Comments* are include as a program code, but it will ignored by compiler and it has differ in term of color. What is the purpose of *Comment* in C programming?
- To increase program readability.
 - To document a program.
 - To provide additional information.
 - As a future references.
- i and ii
 - ii and iii
 - iii and iv
 - i, ii, iii and iv
- Q9 C statements are consisting of;
- (Function, Main, Control, Input/Output, Compound) statements
 - (Function, void, Control, Input/Output, Compound) statements
 - (Function, reserve word, Control, Input/Output, Compound) statements
 - (Function, Declaration, Control, Input/Output, Compound) statements
- Q10 *Reserved words* in C programming contain special meaning understood by compiler, based on the rules below;
- To document a program
 - As a future references
 - Cannot be used as identifier or variables
 - Case –sensitive, must be written in small case
- i and ii
 - ii and iii, iv
 - iii and iv
 - i, ii, iii and iv
- Q11 Which is **not true** about *Identifier*?
- Representing particular name in programming
 - Store values to be used in programming
 - Refers to the storage in computer
 - Refer to the function that will not return any value

- Q12 Select the correct term to declare the constant value for **identifier**?
- #define pi 3.142
 - const float pi = 3.142
 - pi = 3.142
 - pi ==3.142
- i and ii
 - ii and iii
 - iii and iv
 - i, ii, iii and iv
- Q13 Which is true about *data type*?
- Reserved word for integer – int
 - Reserved word for floating number – double / float
 - Reserved word for character – char
 - Reserved word for number – num
- i, ii and iii
 - ii, iii and iv
 - i, iii and iv
 - i, ii, iii, and iv
- Q14 Describe the meaning of expression in C programming?
- Consist of one variable or Constant (operand) which separated by operator.
 - Consist of one variable or Constant (operand) which separated by reserved word.
 - Combination or more than one variable or Constant (operand) which separated by reserved word.
 - Combination or more than one variable or Constant (operand) which separated by operator.
- Q15 Arithmetic Expression is also known as mathematical expression and using arithmetic operator which is divided into two type of operator. List the types of operator?
- Operand and binary operator
 - Proportional and binary operator
 - Unary and binary operator
 - Decimal and binary operator

For questions Q16 and Q17 please refer to figure below.

```

int A, B;
float C;
A= 2;
B= 5;
C= 2.4;
B% A;
C % A;

```

Q16 What is the answer for $B\%A$?

- a. 1
- b. 2
- c. 3
- d. Invalid

Q17 What is the answer for $C\%A$?

- a. 1
- b. 2
- c. 3
- d. Invalid

Q18 Which conditions describe the *double assignment statement*?

- a. total = 80;
- b. int total = sumt = 80
- c. total = sumt++ = 80
- d. int total = 10, sumt = 80

Q19 Which statements are true to describe function of *compound assignment statement*?

- i. To combine two different operator together
 - ii. To simplify arithmetic operator
 - iii. Original function of operator does not affected
 - iv. Allowed the combinations of operator (+=, -=, *=, /=, %=)
- a. i and ii
 - b. ii and iii
 - c. iii and iv
 - d. i, ii, iii and iv

Q20 Which priority of *arithmetic operator precedence rules* will be followed by compiler?

- a. [()] → [++, --] → [*, /, %] → [+, -]
- b. [()] → [*, /, %] → [++, --] → [+, -]
- c. [*, /, %] → [()] → [++, --] → [+, -]
- d. [*, /, %] → [++, --] → [()] → [+, -]

Q21 What is the output for the code below?

```
# include<stdio.h>
# include<math.h>
void main
{
  int a = 25, b;
  b = sqrt(a);
  printf ("%d", b);
}
```

- a. b
- b. 25
- c. 5
- d. invalid

Q22 Which statement is **wrong** about *Relational Expression*?

- a. Combination of more than one statement
- b. Produce 0 if false and 1 if true
- c. Use logical operator
- d. Use relational operator

Q23 Evaluate the following *logical expression*

Given a = 4, b = 2;
! ((a%b<= 5)&&(a-b<3))

- a. 0
- b. 1
- c. 2
- d. 3

Q24 Evaluate the following *logical expression*

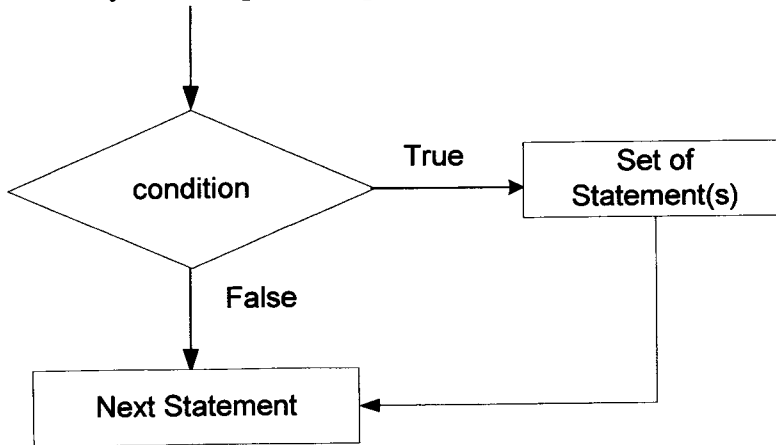
Given a = 4, b = 2;
! ((a%b<= 5) || (a-b<1))

- a. 0
- b. 1
- c. 2
- d. 3

Q25 Following approaches can be chosen as a *control structure*, **except**;

- Sequential approach
- Selectional approach
- Propotional approach
- Iterational approach

Q26 Which syntax is representing the flow chart below?



- ```

if (condition)
{
 Statement(s)
}
Next statement

```
- ```

if (condition)
{
  Statement(s)
}
else
Next statement

```
- ```

if (condition)
{
 Statement set 1
}
else
{
 Statement set 2
}
Next statement

```

```

d. if (condition)
 {
 Statement(s)
 }
 default
 Next statement

```

Q27 Which sentences is true about else if statements

- i. The 'else if' statement is to check for a sequence of conditions
  - ii. When one condition is false, it checks for the next condition and so on
  - iii. When all condition are false the *else* block is executed
  - iv. The statements in that conditional block are executed and the other 'if' statements are skipped
- a. i, ii and iii
  - b. ii, iii and iv
  - c. i, iii and iv
  - d. i, ii, iii, and iv

For questions Q28 and 29 please refer figure below

```

#include<stdio.h>
#include<conio.h>
main()
{
 int result;
 printf("Enter your mark for Test :");
 scanf("%d",&result);
 if(result>=80) {
 printf("Passed: Grade A\n");
 }
 else if (result>=70) {
 printf("Passed: Grade B\n");
 }
 else if (result >=55) {
 printf("Passed: Grade C\n");
 }
 else {
 printf("Failed\n");
 }

 getch();
 return 0;
}

```



Q28 What is the output of the program if the input is 69?

- a. Passed: Grade A
- b. Passed: Grade B
- c. Passed: Grade C
- d. Invalid input

Q29 What is the output of the program if the input is 70.1?

- a. Passed: Grade A
- b. Passed: Grade B
- c. Passed: Grade C
- d. Invalid input

Q30 What is the output of the following code?

```
#include<stdio.h>
#include<conio.h>
int main(void)
{
 int x;
 for(x=1;x<5;x++)
 printf("%d",x);
 getch();
 return 0;
}
```

- a. 1 2 3 4
- b. 2 3 4
- c. 3 4
- d. 4

Q31 What is the function of the *break statement*?

- i. Force the termination of a loop.
  - ii. When a break statement is encountered in a loop, the loop terminates immediately and the execution resumes the next statement following the loop.
  - iii. To produce the systematic code, and the *break statement* doesn't affecting the program.
  - iv. To make the code easy to understand
- a. i and ii
  - b. ii and iii
  - c. ii and iv
  - d. ii

Q32 Which is true about *While* and *do while* loops

- i. *while loop* – the conditions is tested before entering into the loop.
  - ii. *do while loop* – the statements are executed at least once even when the conditions is false.
  - iii. *while loop*- it is an entry- control loop
  - iv. *do while loop* – it is an entry – control loop
- a. i and ii
  - b. ii and iii
  - c. i, ii and iii
  - d. ii, iii and iv

Q33 What type of control structure use by the code below?

```
#include<stdio.h>
#include<conio.h>
int main(void)
{
 int x=1;
 do{
 printf("%d", x++);
 }
 while (x<5);

 getch();
 return 0;
}
```

- a. if
- b. do - while
- c. while
- d. for

Q34 Which function in Programming C that allows you to jump according to designation labels?

- a. go
- b. goto
- c. switch
- d. if-else

Q35 What is switch statement?

- a. a type of decision control structure that selects a choice from the set of available choices
- b. a type of function in programming c that allows you to jump according to designation labels
- c. a type of header
- d. a type of relational expressions