UNIVERSITY OF RUHUNA

Faculty of Engineering

End-Semester 1, Examination, Oct 2022

Module No: EE1101 Module Name: Computer Programming I

Part I

Instructions for candidates

- Write your index number on top of every page.
- Question paper contains 52 multiple choice questions.
- Each question carries 0.5 marks.
- Answer all questions. Each question has only one answer.
- Read the question and all answers before making the choice.
- **For each question, put an X mark on the letter: (a), (b), (c), or (d) which corresponds to the correct answer, by using a black or blue pen.**
- Time allowed is **1 hour and 30 minutes**.

---

1. Founder of the C language is
   (a) Richard Stallman
   (b) Linus Torvalds
   (c) Bill Gates
   (d) Dennis Ritchie
2. Codeblock is a
   (a) Compiler for computer programming languages
   (b) Framework
   (c) NOSQL type Database Management System (DBSM)
   (d) Integrated Development Environment (IDE)
3. A program with a *run-time* error
   (a) can not be executed.
   (b) contains an issue that can not be detected at all.
   (c) contains an issue that is only detected during program execution.
   (d) is syntactically incorrect.
4. A compound statement (a grouped set of statements) is made by using
   (a) ()
   (b) []
   (c) <>
   (d) {}
5. Which of the given words is *not* a keyword in C?
   (a) long
   (b) short
   (c) namespace
   (d) break
6. The code
   `printf("Hello");`
   displays
   (a) `Hello`
   (b) `"Hello"`
   (c) `("Hello")`
   (d) `Hello;`
7. The program with a syntax error is

(a) `int main(void){;}`

(b) `main(void){}`

(c) `int main(void){;;}`

(d) `int main(){ printf("x") return 0;}`

8. The program
```
#include <stdio.h>
int main(void){ printf("ABC"); /*
(:)*/return 0;}
```
(a) prints ABC

(b) prints nothing

(c) has syntax errors

(d) has an unpredictable output

9. If `unsigned int x;`, then

(a) -1 can not be assigned to `x`.

(b) `x/3.4` produces incorrect results.

(c) maximum value that can be assigned to `x` is fixed for a given hardware.

(d) all above answers are correct.

10. Which of the given set of keywords contains only variable types?

(a) `int, string, double`

(b) `continue, unsigned long, short`

(c) `float, long, size`

(d) `unsigned long, short, char`

11. C variable types that best fits to store a kid's name and weight respectively are

(a) array of `float` and `char`.

(b) array of `char` and `float`.

(c) array of `char` and `double`.

(d) array of `pointer` and `void`.

12. C variable type that does not define any particular type is

(a) `float`

(b) `int`

(c) `void`

(d) `char`

13. Which of the variable types is the most suitable for storing the area of a circle?

(a) `double *`

(b) `unsigned double`

(c) `double`

(d) `double &`

14. What is the most efficient method for storing 100 points of Cartesian coordinate system?

(a) `double px[100], py[100];`

(b) `double p[200];`

(c) `struct p[100];`, where `struct p {double x, double y};`

(d) `double *p[100]`

15. If `int x[]={12,13,14},*p=&x[1];` then `printf("%d",*(p+1));` displays

(a) 12

(b) 13

(c) 14

(d) non of the above

16. Which of the format specifiers is used to print the values of a C-String?

(a) %lf

(b) %d

(c) %s

(d) %c

17. Which of the given operators is a binary operator?

(a) %

(b) --

(c) ++

(d) !

18. Which type of operators have the lowest priority?

(a) conditional operators

(b) logical operators

(c) assignment operator

(d) mathematical operators

19. In `#define INTGER 34`, `INTGER` is a

(a) variable

(b) library file

(c) symbolic constant

(d) class

20. What is given by the following statement?
`char name[100];`

    (a) Declaration of array `name` with 100 elements

    (b) Definition of the operator `[ ]`.

    (c) Declaration of a C-string called `name`

    (d) Definition of 100 names called `name`

21. A C-string

    (a) is an array of any type that ends in `'\n'`.

    (b) is an array of type `char` that ends in `'\t'`.

    (c) is an array of type `char` that ends in `'\0'`.

    (d) is an array of type `string`.

22. What is the meaning of `!x != y;` ?

    (a) Is x not equals y?

    (b) Is not-x not-equals y?

    (c) Why y is not equals not-x?

    (d) Is x is not equal to y?

23. What is the meaning of `x = y;` ?

    (a) Assign the value of y to x

    (b) x is equal to y.

    (c) Is x equal to y?

    (d) Is x is not equal to y?

24. What is the meaning of `x == y;` ?

    (a) Value of y is assigned to x.

    (b) x is equal to y

    (c) Is x equal to y?

    (d) Is x is not equal to y?

25. What is the meaning of `x != y;` ?

    (a) Is x is not equal to y?

    (b) Value of y is assigned to x.

    (c) Is x equal to y?

    (d) x is equal to y

26. What does the operation 13%4 produce?

    (a) 0

    (b) 1

    (c) 2

    (d) 3

27. The expression `125 != 124` evaluates to

    (a) 0

    (b) 1

    (c) -1

    (d) 249

28. The expression `1232 == 1232` evaluates to

    (a) 1

    (b) 0

    (c) 6

    (d) 8

29. After execution of
`x=8; ++x; --x;`
the value of x is

    (a) 5

    (b) 6

    (c) 7

    (d) 8

30. After execution of
`x=15; x += x;`
the value of x is

    (a) 15

    (b) 16

    (c) 225

    (d) 30

31. The expression `(x + y * 5) < 25` evaluates to 1 if

    (a) x = 3 and y = 4

    (b) x = 0 and y = 5

    (c) x = 11 and y = 4

    (d) x = 14 and y = 3

32. The expression `((4+40) == 44 || 3 != 3 )` evaluates to

    (a) 33

    (b) 42

    (c) 1

    (d) 0

33. If `y=10;` then the expression
`(y -= 5) && ((45+10)>(5*9))`
evaluates to

    (a) 1
    (b) 0
    (c) -1
    (d) Non of the above choices

34. If `x='4';` then the expression
`!(x>='a' && x<='z')`
evaluates to

    (a) 0
    (b) 1
    (c) '4'
    (d) 134

35. `if(!x) printf("Yes");`
This displays `Yes` only if `x` is

    (a) greater than 0 or less than 0.
    (b) is less than 0.
    (c) greater than 0.
    (d) equal to 0.

36. `if( x+'A' > 'A') printf("Yes");`
This displays `Yes` only if `x` is

    (a) greater than 'A'.
    (b) less than 0.
    (c) greater than or equal to 0.
    (d) greater than 0.

37. `if( x-25 < 25) printf("Yes");`
This displays `Yes` if `x` is

    (a) greater than 50.
    (b) greater than 10.
    (c) less than 70.
    (d) less than 50.

38. `if( x > 1010 && x <=2020) printf("Yes");`
`else printf("No");`
This displays `No` if `x` is

    (a) 1122
    (b) 1001
    (c) 1105
    (d) 2020

39. `if(x<=150 || x>175) printf("Yes");`
This displays `Yes` if `x` is

    (a) 150
    (b) 160
    (c) 170
    (d) 175

40. `if((x<10 || x>25) && (x<-10 || x>-25))`
`printf("Yes");`
This displays `Yes` if `x` is

    (a) -15
    (b) 0
    (c) -30
    (d) any of the above.

41. `(12==5 && 3!=3) || (4+5 || 3-4+1)`
This expression evaluates to

    (a) -1
    (b) 0
    (c) 1
    (d) non of the above values.

42. `for(i=0;i<611;++i) printf("X");`
How many times the character `X` is displayed?

    (a) 610
    (b) 611
    (c) 612
    (d) 0

43. `for(i=0;i<=12; i += 2) printf("X");`
How many times the character `X` is displayed?

    (a) 5
    (b) 6
    (c) 7
    (d) 8

44. `for(i=5,j=71;i>-5; i-=2, j+=3) printf("%d",j);`
What is displayed?

    (a) 7275788184
    (b) 7174778184
    (c) 7174778083
    (d) 7073747982

45. `i=5;while(i<10){printf("%d",i);++i;}`
This code prints

    (a) 56789
    (b) 5678910
    (c) 45678
    (d) 45678910

46. `i=0;while(i<4)++i;printf("%d",i);`
The value of `i` displayed is

    (a) 0123
    (b) 4
    (c) 3210
    (d) Non of the above

47. `Sum=4;i=2;while(i<=5){Sum+=i;++i;}`
This code sets `Sum` to

    (a) 17
    (b) 16
    (c) 15
    (d) 18

48. In `switch` statement

    (a) `goto` can be used to direct to another case.
    (b) `default` is not optional.
    (c) `break` prevents execution of next `case`.
    (d) `continue` makes execution of previous `case`.

49.
```
for(i=-5, j=11;i<34; i+=j,--j)
{  printf(" %d ", j); }
```
Above code displays

    (a) 11 10 9 8 7
    (b) 11 9 7 5 3
    (c) 10 8 8 6 -5
    (d) 10 9 5 -3 2

50.
```
i=1000;
while(i)
{  i-=2;
   printf("\n%d", i);
}
```
Above code displays

    (a) odd numbers.
    (b) even numbers.
    (c) fractional numbers.
    (d) prime numbers.

51. `void tow(float *x);`
is a function

    (a) definition
    (b) call
    (c) name
    (d) prototype

52. According to
`double xy(int x, float y);`
the return value is

    (a) of type `float`.
    (b) of type `int`.
    (c) of type `double`.
    (d) of non of the types given above.