Advanced C Programming Test

Write this with your college mail Id.

* Indicates required question

1. Email *

2. Name *

3. ID(S20....) *

4. select your section *

Mark only one oval.

○ 3A

○ 3B

○ 3C

○ 3D

○ 3E

○ 3F

Untitled Section

TAKE YOUR TEST

5. 1.What will be output if you will compile and execute the following c code? * 1 point

```
void main(){
int a,b;
a=1,3,15;
b=(2,4,6);
clrscr();
printf("%d ",a+b);
getch();
}
```

Mark only one oval.

○ 3

○ 21

○ 17

○ 7

○ compiler error

6. 2.What will be output if you will compile and execute the following c code? * 1 point

```
void main(){
if(printf("cquestionbank"))
printf("I know c");
else
printf("I know c++");
}
```

Mark only one oval.

○ I know c

○ I know c++

○ cquestionbankI know c

○ cquestionbankI know c++

○ compiler error

7. 3.#include <stdio.h> * 1 point

```
int main()
{
    int i = 5, j = 10, k = 15;
    printf("%d ", sizeof(k /= i + j));
    printf("%d", k);
    return 0;
}
```

Assume size of an integer as 4 bytes. What is the output of above program?

Mark only one oval.

○ 4 1

○ 4 15

○ 2 1

○ compiler error

8. 4.The following three \'C\' language statements is equivalent to which single statement? y=y+1; z=x+y; x=x+1 * 1 point

Mark only one oval.

○ z=(x++)+(++y);

○ z=(x++)+(y++);

○ z=(x++)+(++y)+1;

○ z=x+y+2;

9. 5.In the following recursive function, how many times is the function f called when f(5) is executed? * 1 point

```
void f(int n) {
    if (n <= 0)
        return;
    f(n - 1);
    f(n - 2);
}
```

Mark only one oval.

○ 8

○ 15

○ 16

○ 10

10.

```
6.struct marks{
int p:3;
int c:3;
int m:2;
};
void main(){
struct marks s={2,-6,5};
printf("%d %d %d",s.p,s.c,s.m);
}
```

* 1 point

Mark only one oval.

○ 2 -6 5

○ 2 -6 1

○ 2 2 1

○ compiler error

11. 7.Which code from the given option return pointer to last occurrence of c in ch or NULL if not present? * 1 point

Mark only one oval.

○ char*strchr(ch,c)

○ char*strrchr(ch,c)

○ char*strncat(ch,c)

○ chr*strcat(ch,c)

12. 8.#include<stdio.h> * 1 point

```
int main()
{
    int *p = (int *)malloc(sizeof(int));

    p = NULL;

    free(p);
}
```

Mark only one oval.

○ compiler error:free cant be applied on NULL pointer

○ memory leak

○ dangling pointer

○ the program may crash as free() is called for NULL pointer

13. 9.predict the output of the below code * 1 point

```
#include<stdio.h>
void main(){
int a=2;
switch(a)
{
case 1:printf("CSE");
case 2:
continue;
case 3:
printf("CODER");
}
}
```

Mark only one oval.

○ error

○ CSE

○ CODER

○ executed succesfully with empty screen

14. 10.Which of the following is a key advantage of recursion over iteration? * 1 point

A) Reduced execution time

B) Ability to solve problems that cannot be broken into subproblems

C) Better space utilization

D) Simpler code for problems like tree traversal

Mark only one oval.

○ A

○ B

○ C

○ D

15. 11.#include <stdio.h> * 1 point

```
int main()
{
    int i = 97, *p = &i;
    foo(&i);
    printf("%d ", *p);
}
void foo(int *p)
{
    int j = 2;
    p = &j;
    printf("%d ", *p);
}
```

Mark only one oval.

○ 2 97

○ 2 2

○ compiler error

○ segmentation fault code dumped or code crashed

16. 12.The most appropriate matching for the following pairs * 1 point

X: m=malloc(5); m= NULL; 1: using dangling pointers
Y: free(n); n->value=5; 2: using uninitialized pointers
Z: char *p; *p = ’a’; 3. lost memory

Mark only one oval.

○ X-1 ,Y-3 ,Z-2

○ X-2,Y-1,Z-3

○ X-3,Y-2,Z-1

○ X-3,Y-1,Z-2

17. 13.What will be output if you will compile and execute the following c code? * 1 point

```
void main(){
static main;
int x;
x=call(main);
clrscr();
printf("%d ",x);
getch();
}
int call(int address){
address++;
return address;
}
```

Mark only one oval.

○ 0

○ 1

○ garbage value

○ compiler error

18. 14.#include<stdio.h> * 1 point

```
void main()
{
int a=36;
int b=9;
printf("%d",a>>a/b-2);
}
```

Mark only one oval.

○ 9

○ 7

○ 5

○ none of these

19. 15.Which of the following is true? * 1 point

Mark only one oval.

○ gets() doesn\'t do any array bound testing and should not be used.

○ fgets() should be used in place of gets() only for files, otherwise gets() is fine

○ gets() cannot read strings with spaces

○ None of the above

20. 16.#include<stdio.h> * 1 point

```
float i=2.0;
float j=1.0;
float sum = 0.0;
main()
{
    while (i/j > 0.001)
    {
        j+=j;
        sum=sum+(i/j);
        printf(\"%f\\n\", sum);
    }
}
```

Mark only one oval.

○ 9

○ 8

○ 10

○ 11

21. 17. void f(int* p, int m) *

1 point

```
{
    m = m + 5;
    *p = *p + m;
    return;
}
void main()
{
    int i=5, j=10;
    f(&i, j);
    printf(\"%d\", i+j);
}
```

Mark only one oval.

○ 10

○ 20

○ 30

○ 40

22. 18.int fun() *

1 point

```
{
  static int num = 16;
  return num--;
}

int main()
{
  for(fun(); fun(); fun())
    printf("%d ", fun());
  return 0;
}
```

Mark only one oval.

○ 13 10 7 4 1

○ 15 12 8 5 2

○ 14 11 8 5 2

○ infinite loop

23. 19.How many times TNPC is printed? * 1 point

```
int main()
{
int a = 0;
while(a++ < 5-++a)
printf("TNPC");
return 0;
}
```

Mark only one oval.

○ 2 times

○ 3 times

○ 1 time

○ 4 times

24. Which gcc flag is used to genarate debug information for any binary file? * 1 point

Mark only one oval.

○ gcc -g

○ gcc -a

○ gcc -e

○ gcc -b