1)

void main(){

   int i=10;

   static int x=i;

   if(x==i)

      printf("Equal");

   else if(x>i)

      printf("Greater than");

   else

      printf("Less than");

}

(a) Equal

(b) Greater than

(c) Less than

(d) Compiler error

Answer: (d)

*2)*

*main()*

*{*

*static int var = 5;*

*printf("%d ",var--);*

*if(var)*

*main();*

*}*

*a)5 4 3 2 1 b)5 5 5 5 5 c)5 d)error*

***Answer:***

5 4 3 2 1

3)

void main(){

   printf("%s","c" "question" "bank");

}

(a) c question bank

(b) c

(c) bank

(d) cquestionbank

(e) Compiler error

Answer: (d)

4)

*main()*

*{*

*int i=5;*

*printf("%d");*

*}*

*a)error b)0 c)5 d)garbage value*

*ans: c*

5)

void main(){

   char \*str="c-pointer";

   printf("%\*.\*s",10,7,str);

}

(a) c-pointer

(b) c-pointer

(c) c-point

(d) cpointer null null

Answer: (c)

6)

*main()*

*{*

*extern int i;*

*i=20;*

*printf("%d",i);*

*}*

*a)20 b)0 c)linker error d)garbage value*

*ans:c)*

7)

void start();

void end();

#pragma startup start

#pragma exit end

int static i;

void main(){

   printf("\nmain function: %d",++i);

}

void start(){

   clrscr();

   printf("\nstart function: %d",++i);

}

void end(){

   printf("\nend function: %d",++i);

   getch();

}

(a)

main function: 2

start function: 1

end function:3

(b)

start function: 1

main function: 2

end function:3

(c)

main function: 2

end function:3

start function: 1

(d) Compiler error

Answer: (b)

*8)*

main( )

{

static int a[ ] = {0,1,2,3,4};

int \*p[ ] = {a,a+1,a+2,a+3,a+4};

int \*\*ptr = p;

ptr++;

printf(“\n %d %d %d”, ptr-p, \*ptr-a, \*\*ptr);

\*ptr++;

printf(“\n %d %d %d”, ptr-p, \*ptr-a, \*\*ptr);

\*++ptr;

printf(“\n %d %d %d”, ptr-p, \*ptr-a, \*\*ptr);

++\*ptr;

printf(“\n %d %d %d”, ptr-p, \*ptr-a, \*\*ptr);

}

***a)*** *111 b) 111 c) 111 d) 111*

*222 222 222 111*

*333 111 333 333*

*344 222 444 444*

*Ans: a*

9)

void main(){

   int a,b;

   a=1,3,15;

   b=(2,4,6);

   clrscr();

   printf("%d ",a+b);

   getch();

}

(a) 3

(b) 21

(c) 17

(d) 7

Answer: (d)

*10)*

*main()*

*{*

*char string[]="Hello World";*

*display(string);*

*}*

*void display(char \*string)*

*{*

*printf("%s",string);*

*}*

***a)error b)Hello World c)Hello d)no output***

***ans:a)***

11)

#define message "union is\

power of c"

void main(){

   clrscr();

   printf("%s",message);

   getch();

}

(a) union is power of c

(b) union ispower of c

(c) union is

Power of c

(d) Compiler error

Answer: (b)

12)

char \*strexp()

         {

         char \*temp = "example string";

         return temp;

         }

         int main()

         {

         puts(strexp);

         }

a)example string b)garbage value c)error d)no output

ans:example string

|  |
| --- |
| 13)  include<stdio.h>  void f(int \*p, int \*q)  {  p = q;  \*p = 2;  }  int i = 0, j = 1;  int main()  {  f(&i, &j);  printf("%d %d \n", i, j);  getchar();  return 0;  } |
|  |

1. 2 2  
   (B) 2 1  
   (C) 0 1  
   (D) 0 2

Answer (D)

***14)***

*main()*

*{*

*int i=5;*

*printf("%d%d%d%d%d%d",i++,i--,++i,--i,i);*

*}*

*a)45545 b)56655 c)65655 d)65545*

ans:a

1. main()

{

Char \*p=”algc”;

Printf(“%c”,++\*(p++));

Printf(“%c”,\*++p);

}

al b) bg c) lg d) none of the above

ans b

**16) Consider the following C-function in which a[n] and b[m] are two sorted integer arrays and c[n + m] be another integer array.**

|  |
| --- |
| void xyz(int a[], int b [], int c[])  {    int i, j, k;    i = j = k = O;    while ((i<n) && (j<m))       if (a[i] < b[j]) c[k++] = a[i++];       else c[k++] = b[j++];  } |

**Which of the following condition(s) hold(s) after the termination of the while loop?  
(i) j < m, k = n+j-1, and a[n-1] < b[j] if i = n  
(ii) i < n, k = m+i-1, and b[m-1] <= a[i] if j = m**  
(A) only (i)  
(B) only (ii)  
(C) either (i) or (ii) but not both  
(D) neither (i) nor (ii)

Answer (C)

|  |
| --- |
| 17)  int a, b, c = 0;  void prtFun (void);  int main ()  {      static int a = 1; /\* line 1 \*/      prtFun();      a += 1;      prtFun();      printf ( "\n %d %d " , a, b) ;  }    void prtFun (void)  {      static int a = 2; /\* line 2 \*/      int b = 1;      a += ++b;      printf (" \n %d %d " , a, b);  } |

**What output will be generated by the given code segment?**  
(A) 3 1  
4 1  
4 2  
(B) 4 2  
6 1  
6 1  
(C) 4 2  
6 2  
2 0  
(D) 3 1  
5 2  
5 2

Answer (C)

*18)*

*#include <stdio.h>*

*#define a 10*

*main()*

*{*

*#define a 50*

*printf("%d",a);*

*}*

*a)10 b)50 c)0 d)error*

*ans:b*

**19) Consider the following recursive C function that takes two arguments**

|  |
| --- |
| unsigned int foo(unsigned int n, unsigned int r) {    if (n  > 0) return (n%r +  foo (n/r, r ));    else return 0;  } |

**What is the return value of the function foo when it is called as foo(513, 2)?**  
(A) 9  
(B) 8  
(C) 5  
(D) 2

Answer: (D)

20)

#include<stdarg.h>

int ripple ( int , ...);

main()

{

int num;

num = ripple ( 3, 5,7);

printf( " %d" , num);

}

int ripple (int n, ...)

{

int i , j;

int k;

va\_list p;

k= 0;

j = 1;

va\_start( p , n);

for (; j<n; ++j)

{

i = va\_arg( p , int);

for (; i; i &=i-1 )

++k;

}

return k;

}

The output for this program is:   
  
(a) 7  
(b) 6  
(c) 5  
(d) 3

The answer is (c)