1.
```
#include<stdio.h>
#define sunbeam 1998
int main( void )
{
    #ifndef sunbeam
        printf("sunbeam");
    #endif
    printf("SunBeam");

    return 0;
}
```

A. SunBeam
B. sunbeam
C. 1998
d. sunbeam 1998

Answer: A

2.
```
#include<stdio.h>
#include<stdlib.h>
int main( void )
{
   int *a[3];
   a = (int*) malloc(sizeof(int)*3);
   free(a);
   return 0;
}
```

A. unable to allocate memory
B. compile time error as incompatible types
C. unable to free memory
D. no error

Answer : B

3.
if input is 9 9 9 9

```
#include<stdio.h>
#define max 100
int main( void )
{
    int i,sum=0;
    int *ptr = (int*) malloc(1 * sizeof(int));

    printf("Enter elements: ");
    for(i = 0; i < 4; ++i)
    {
         scanf("%d", ptr + i);
         sum += *(ptr+1);
    }
    printf("Sum = %d", sum);

    return 0;
}
```

A. 18
B. 27
C. 36
D. 9

Answer: B

4.
C Processor

A. Takes care of conditional compilation
B. Takes care of macros
C. Takes care of include files
D. All of the above

Answer: D

5.
```
#include<stdio.h>
int main( void )
{
    int *sum = (int *)malloc(sizeof(int));
    sum = NULL;
    free(sum);
    return 0;
}
```

A. Compliation Error
B. Error free
C. Memory Leakage
D. dangling pointer

Answer : C

6.
```
#include<stdio.h>
#define max 100
int main( void )
{
    #ifndef max
        #ifdef max
            if(max)
                printf("Sunbeam");
            #endif
        printf("Pune");
    #endif
    return 0;
}
```

A. Pune
B. sunbeam
C. sunbeamPune
D. no output
Answer: D

7.
```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main( void )
{
    char * ch=NULL;
    ch=malloc(20);
    strcpy(ch,"sunbeam");
    printf("%s",ch);
    ch=realloc(ch,0);
    printf("%s",ch);
    return 0;
}
```

A. sunbeam(null)
B. sunbeam
C. Run time error
D. Compile time error

Answer: A

8.
```
#include<stdio.h>
#define print(Y,X) (Y/Y,X*Y)
int main( void )
{
    printf("%d",print(5,9));
    return 0;
}
```

A. 1
B. 81
C. 45
D. 0

Answer: C

```
9.
#include<stdio.h>
#define X(Y) (Y > Y)
#define Y(X) (X <= X)
int main( void )
{
    int sum=X(4) + Y(3);
    printf("%d",sum);

    return 0;
}

A. 1
B. 0
C. 4
D. 3
Answer : A

10.
#include <stdio.h>
#define int float*
int main( void )
{
    int j=NULL, i=50;

    printf("\t sizeof (i) =%d", sizeof(i));
    printf("\t sizeof (j) =%d", sizeof(j));

    return 0;
}    // Note : Consider 64 bit compliation

A. sizeof (i) = 4   sizeof (j) = 4
B. sizeof (i) = 8   sizeof (j) = 8
C. sizeof (i) = 4   sizeof (j) = 8
D. sizeof (i) = 2   sizeof (j) = 4

Answer : C
```

11.
```
#include<stdio.h>
#define sum(Y,X) Y>X?Y>X:X>Y
int main( void )
{
    int a= sum(2,3);
    printf("%d",a);

    return 0;
}
```

A. 0
B. 2
C. 3
D. 1

Answer : D

12.
```
#include <stdio.h>
#define print() printf("%d ",10/2)
int main( void )
{
    printf("%d",print());

    return 0;
}
```

A. 10 1
B. 2 2
C. 5 1
D. 5 2

Answer : D

```
13.
#include <stdio.h>
#define min(X,Y) (X>Y)?printf("%d",X):printf("%d" ,Y)
int main( void )
{
    printf(" %d ",min(11,11));

    return 0;
}
A. 11 2
B. 11 1
C. 11 11
D. none
Answer : A

14.
#include <stdio.h>
struct emp
{
    struct emp *next;
    int sal;
};
int main(void)
{
    struct emp *p1 = calloc(1, sizeof(struct emp));
    p1->sal = 1;
    p1->next = calloc(1, sizeof(struct emp));
     printf("%d\n", p1->next->sal);
     return 0;
}

A. 0
B. 4
C. 8
D. 2

Answer: A
```

15.
```
#include<stdlib.h>
int main(void)
{
    char ptr[]="PG-DMC";

    strcpy(ptr , "Demo1");
    strcpy(ptr , "Demo2");
    free(ptr);

    return 0;
}
```

A. Demo1
B. Demo2
C. Demo1Demo2
D. exit value -1

Answer: D

16.
what type of data u can store in this block of memory?
```
#include<stdio.h>
#include<stdlib.h>
int main(void)
{
    void *ptr=NULL;
    ptr = malloc(10);
    return 0;
}
```
A. int
B. char
C. float
D. all of above data types

Answer: D

17.
Which of the above three functions are likely to cause problems with pointers?

```
int * fun1 (void)
{
  int x= 10;
  return (&x);
}
int * fun2 (void)
{
  int * px;
  *px= 10;
  return px;
}
int *fun3 (void)
{
  int *px;
  px = (int *) malloc (sizeof(int));
  *px= 10;
  return px;
}
```

A. function fun1 and fun2
B. function fun2 and fun3
C. function fun1 , fun2 and fun3
D. function fun3

Answer: A