```
/*Write a C Program using Dynamic Memory Allocation for the following problem
statements
1. to create memory for int, char and float variable at run time. */
#include<stdio.h>
#include<stdlib.h>
main()
{
 int *p;
 char *c;
 float *f;
 p=(int*)malloc(sizeof(int));
 c=(char*)malloc(sizeof(char));
 f=(float*)malloc(sizeof(float));
 printf("Enter a int , char and float value : ");
 scanf("%d %c %f",p,c,f);
 printf("int = %d, char = %c, float = %f",*p,*c,*f);
 free(p);
 free(c);
 free(f);
 getch();
 return 0;
}
```

OUTPUT

```
Enter a int, char and float value:45
p
5.91
Int=45,char=p,float=5.910000
```

```
/*Write a C Program using Dynamic Memory Allocation for the following problem
statements
2. to input and print text using Dynamic Memory Allocation. */
main()
{
 char *p;
 p=(char*)malloc(50);
 printf("Enter a string : ");
 gets(p);
 printf("%s",p);
 getch();
 return 0;
}
```

OUTPUT

```
Enter a string : hy I am Pallabi sethi
Hy I am Pallabi sethi
```

```
/*Write a C Program using Dynamic Memory Allocation for the following problem
statements
3. to read a one dimensional array, print sum of all elements along with inputted array
elements using Dynamic Memory Allocation. */
main()
{
 int *p,sum=0,size;
 printf("Enter size : ");
 scanf("%d",&size);

 p=(int*)calloc(size,4);

 printf("Enter %d elements : ",size);

 for(int i=0; i<size; i++)

 scanf("%d",p+i);

 printf("Array is \n");

 for(int i=0; i<size; i++)

 {

 sum+=*(p+i);

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

 }

 printf("\nSum of all elements are : %d",sum);

 getch();

 return 0;

}
```

OUTPUT

```
 Enter size:6

Enter 6 elements:65

87

22

44

99

98

Array is:65 87 22 44 99 98

Sum of all elements are:414
```

```
/*Write a C Program using Dynamic Memory Allocation for the following problem
statements
4. to read and print the student details using structure and Dynamic Memory Allocation. */
typedef struct
{
 char name[20];
 int rollno;
}student;
main()
{
 student *p;
 p=(student*)malloc(sizeof(student));
 printf("Enter name of the student : ");
 gets(p->name);
 printf("Enter roll no of %s : ",p->name);
 scanf("%d",&p->rollno);
 printf("Name : %s, Roll No : %d",p->name,p->rollno);
 getch();
 return 0;
}
```

OUTPUT

```
Enter name of the student : Pallabi
Enter Roll No of Pallabi : 100
Name : pallabi ,Roll No : 100
```

```
/*Write a C Program using Dynamic Memory Allocation for the following problem
statements
5. to find sum of N elements entered by user. To perform this program, allocate memory
dynamically using malloc() function. */
main()
{
 int *p,sum=0,n;
 printf("Enter the value of n : ");
 scanf("%d",&n);
 p=(int*)malloc(sizeof(int)*n);
 printf("Enter %d elements : ",n);
 for(int i=0; i<n; i++)
 scanf("%d",p+i);
 printf("Elements Are : \n");
 for(int i=0; i<n; i++)
 {
 sum+=*(p+i);
 printf("%d ",*(p+i));
 }
 printf("\nSum of %d elements is : %d",n,sum);
 getch();
 return 0;
}
```

OUTPUT

```
Enter the value of n : 6
Enter 6 elements:65
87
22
44
99
```

```
98
Elements are : 65 87 22 44 99 98
Sum of 6 elements is:414
```

```
/*Write a C Program using Dynamic Memory Allocation for the following problem
statements
6. to find Largest of N Numbers. To perform this program, allocate memory dynamically
using calloc() and realloc() function.
*/
main()
{
 int *p;
 p=(int*)malloc(4);
 int n;
 printf("Enter value of n : ");
 scanf("%d",&n);
 p=(int*)realloc(p,sizeof(int)*n);
 printf("Enter %d numbers : ",n);
 for(int i=0; i<n; i++)
 scanf("%d",p+i);
 int large=*p;
 for(int i=1; i<n; i++)
 {
 if(large<*(p+i))
 large=*(p+i);
 }
 printf("Largest number : %d",large);
 getch();
 return 0;
}
```

OUTPUT

```
Enter the value of n : 6

Enter 6 elements:65

87

22

44

99

98

Largest number : 99
```

```
/* Write a C Program using Pre-processors for the following problem statements

7. Display all prime numbers between two Intervals

*/

#define START 1

#define END 100

main()

{

 #ifdef START && END

 for(int n=START; n<=END; n++)

 {

 int i;

 for(i=2; i<n; i++)

 if(n%i==0)

 break;

 if(i==n)

 printf("%d ",n);

 }
 #endif // START
}
```

OUTPUT

```
2 3 5 7 11 19 23 29 31 37b41 43 47 53 59 61 67 71 73 73 83 89 97
```

```
/* Write a C Program using Pre-processors for the following problem statements
8. Check Prime and Armstrong Number by making function */
#define NUM 153
main()
{
 #ifdef NUM
 prime(NUM);
 armstrong(NUM);
 #endif // NUM
}
void prime(int n)
{
 int i;
 for(i=2; i<n; i++)
 if(n%2==0)
 break;
 if(i==n)
 printf("%d is prime\n",n);
 else
 printf("%d is not prime\n",n);
}
void armstrong(int n)
{
 int r,num,t;
 t=n;
 while(n)
 {
 num=num+(pow((n%10),3));
 n=n/10;
 }
 if(num==t)
 printf("%d is armstrong",num);
 else
 printf("%d is not a armstrong number ",t);
}
```

OUTPUT

```
153 is prime
153 is armstrong
```

```
/* Write a C Program using Pre-processors for the following problem statements
9. Define a preprocessor macro swap(t, x, y) that will swap two arguments x and y of a given
type t. */
#define swap(t,x,y){t temp; temp=x; x=y; y=temp;}
main()
{
 printf("Enter 2 numbers : ");
 int a,b;
 scanf("%d %d",&a,&b);
 swap(int,a,b);
 printf("%d %d",a,b);
 getch();
 return 0;
}
```

OUTPUT

```
Enter 2 number : 2 3
2 3
```

```
/* 10.Define a preprocessor macro to select:
o the least significant bit from an unsigned char
o the nth (assuming least significant is 0) bit from an unsigned char.
*/
#define LSB(x) (x = 0x01)
#define LSB1(x) (x&1)
#define nSB(x,n) (x&(1<<n))
#define lsb(a) a%2
main()
{
 unsigned char a='A';
 printf("%d\n",nSB(a,3));
 printf("%d",LSB(a));
}
```

OUTPUT

```
0
1
```