Assignment – 22

DMA

1. Define a function to input variable length string and store it in an array without

memory wastage?  
solve –

#include<stdio.h>

int main()

{

char \*str, c;

int i=0,j=0;

str =(char\*)malloc(sizeof(char));

printf("Enter the string := ");

while( c != '\n')

{

c = getc(stdin);

j++;

str = (char\*)realloc(str,j \* sizeof(char));

str[i]= c;

i++;

}

str[i]= '\0';

printf(" entered string is %s",str);

free(str);

return 0;

}

Output -

Enter the string := sagar

entered string is sagar

2. Write a program to ask the user to input a number of data values he would like to

enter then create an array dynamically to accommodate the data values. Now take

the input from the user and display the average of data values ?  
solve -

#include<stdio.h>

int main()h

{

int \*ptr , c;

int i=0 , size = 0 , sum = 0;

printf("Enter the size of array \n");

scanf("%d\n",&size);

ptr = (int\*)calloc(size,sizeof(int));

if(ptr==NULL)

{

printf("memory allocation failed \n");

return 0;

}

printf("entered %d\n value is \n",size);

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

{

scanf("%d\n",ptr+i);

}

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

{

sum = sum + \*(ptr + i);

}

printf(" the average is %d\n",sum/size);

free(ptr);

return 0;

}

Output –

Enter the size of array

5

entered 5

value is

1

2

3

4

5

the average is 3

3. Write a program to calculate the sum of n numbers entered by the user using malloc

and free.

Solve -

#include<stdio.h>

int main()

{

int \*ptr,size = 0;

int i=0, sum = 0 ;

printf("Enter the size of array \n");

scanf("%d\n",&size);

ptr = (int\*)malloc(size \* sizeof(int));

if(ptr==NULL)

{

printf("memory allocation is failed\n");

return 0;

}

printf("enter the value of array \n");

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

{

scanf("%d\n",ptr+i);

}

printf(" entered %d value is \n",size);

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

{

sum = sum + \*(ptr + i);

}

printf(" the sum of n number is %d\n",sum);

free(ptr);

return 0;

}

Output -

Enter the size of array

5

enter the value of array

1

2

3

4

5

entered 5 value is

the sum of n number is 15

4. Write a program to input and print text using dynamic memory allocation ?

Solve –

#include<stdio.h>

int main()

{

char \*str, c;

int i=0,j=0;

str =(char\*)malloc(sizeof(char));

printf("Enter the string := ");

while( c != '\n')

{

c = getc(stdin);

j++;

str = (char\*)realloc(str,j \* sizeof(char));

str[i]= c;

i++;

}

str[i]= '\0';

printf(" entered string is %s",str);

free(str);

return 0;

}

Output -

Enter the string := sagar

entered string is sagar

5. Write a program to read a one dimensional array, print sum of all elements along with

inputted array elements using dynamic memory allocation ?

solve -

#include<stdio.h>

int main()

{

int \*ptr,size = 0;

int i=0, sum = 0 ;

printf("Enter the size of array \n");

scanf("%d\n",&size);

ptr = (int\*)malloc(size \* sizeof(int));

if(ptr==NULL)

{

printf("memory allocation is failed\n");

return 0;

}

printf("enter the value of array \n");

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

{

scanf("%d\n",ptr+i);

}

printf(" entered %d value is \n",size);

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

{

sum = sum + \*(ptr + i);

}

printf(" the sum of n number is %d\n",sum);

free(ptr);

return 0;

}

Output -

Enter the size of array

4

enter the value of array

1

2

3

4

entered 4 value is

the sum of n number is 10

6. Write a program in C to find the largest element using Dynamic Memory Allocation ?

Solve –

#include<stdio.h>

int main()

{

int \*ptr,size = 0;

int i, sum = 0 , max = -9999999;

printf("Enter the size of array \n");

scanf("%d\n",&size);

ptr = (int\*)malloc(size \* sizeof(int));

if(ptr==NULL)

{

printf("memory allocation is failed\n");

return 0;

}

printf("enter the value of array \n");

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

{

scanf("%d\n",(ptr+i));

}

printf(" entered %d value is \n",size);

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

{

if(max < \*(ptr+i))

max = \*(ptr+i);

}

printf(" the maximum value is %d\n",max);

free(ptr);

return 0;

}

Output -

Enter the size of array

5

5

enter the value of array

12

13

41

23

32

entered 5 value is

the maximum value is 41

7. Write a program to demonstrate memory leak in C ?

Solve -

#include<stdio.h>

int main()

{

int \*ptr ;

ptr = (int\*)malloc(sizeof(int));

ptr = NULL;

return 0;

}

10. Find out the maximum and minimum from an array using dynamic memory allocation

in C ?

solve -

#include<stdio.h>

int main()

{

int \*ptr,size = 0;

int i, sum = 0 , max = -9999999 , min = 999999;

printf("Enter the size of array \n");

scanf("%d\n",&size);

ptr = (int\*)malloc(size \* sizeof(int));

if(ptr==NULL)

{

printf("memory allocation is failed\n");

return 0;

}

printf("enter the value of array \n");

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

{

scanf("%d\n",(ptr+i));

}

printf(" entered %d value is \n",size);

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

{

if(max < \*(ptr+i))

max = \*(ptr+i);

}

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

{

if( min > \*(ptr+i))

min = \*(ptr+i);

}

printf(" the maximum value is %d\n",max);

printf(" the minimum value is %d\n",min);

free(ptr);

return 0;

}

Output -

Enter the size of array

3

3

enter the value of array

1

56

4

entered 3 value is

the maximum value is 56

the minimum value is 1

8. Write a program to demonstrate dangling pointers in C ?

Solve -

#include<stdio.h>

int main()

{

int \*ptr ;

ptr = (int\*)malloc(sizeof(int));

\*ptr = 10 ;

printf(" before free %d\n",\*ptr);

free(ptr);

printf(" after free %d\n",\*ptr);

return 0;

}

Output -

before free 10

after free 13308536

7. Write a program to demonstrate memory leak in C ?  
solve -

#include<stdio.h>

int main()

{

int \*ptr ;

ptr = (int\*)malloc(sizeof(int));

ptr = NULL;

return 0;

}

9. Write a program to allocate memory dynamically of the size in bytes entered by the

user. Also handle the case when memory allocation is failed ?

solve –

#include<stdio.h>

int main()

{

int \*ptr;

ptr = (int\*)malloc(sizeof(int));

if(ptr==NULL)

{

printf("memory allocation is failed");

}

return 0;

}