Assignment 22

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

memory wastage.

Input;

#include<iostream>

int main()

{

    char \*str,c;

    int i=0,j=1;

    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("the entered string is %s",str);

    free(str);

    printf("\nthe  size of string is: %d ", sizeof(str));

    return 0;

}

Output:

enter the string: tushar

the entered string is tushar

the size of string is: 4

PS C:\Users\tusha\Documents\coadind>

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.

Input:

#include<iostream>

int main()

{

    int \*marks,size,sum=0;

    printf("enter teh size of array:");

    scanf("%d",&size);

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

    if(marks==NULL)

    {

        printf("memory allocation failed.");

        return 0;

    }

    printf("enter the values: \n");

    for (int i = 0; i < size; i++)

    {

        scanf("%d",marks+i);

    }

    for (int i = 0; i < size; i++)

    {

        sum=sum + \*(marks+i);

    }

    printf("the sum is %d",sum);

    printf("\nthe average of the number is %d",(sum/size));

    return 0;

}

Output:

enter teh size of array:4

enter the values:

10 20 30 40

the sum is 100

the average of the number is 25

PS C:\Users\tusha\Documents\coadind>

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

and free.

Input:

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

//and free.

#include<iostream>

int main()

{

    int \*ptr,size,sum=0;

    printf("enter the size :");

    scanf("%d",&size);

    ptr=(int\*)malloc(sizeof(int));

    if(ptr==NULL)

    {

        printf("memory allocation is failed.");

        return 0;

    }

    printf("enter the numbers:\n");

    for (int i = 0; i < size; i++)

    {

        scanf("%d",ptr+i);

    }

    for (int i = 0; i < size; i++)

    {

        sum=sum + \*(ptr+i);

    }

    printf("the sum of the entered value is : %d",sum);

    free(ptr);

    return 0;

}

Output:

enter the size :5

enter the numbers:

1 2 3 4 5

the sum of the entered value is : 15

PS C:\Users\tusha\Documents\coadind>

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

inputted array elements using dynamic memory allocation.

#include<iostream>

int main()

{

    int \*ptr,size,sum=0;

    printf("enter the size of the array.");

    scanf("%d",&size);

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

    printf("enter the numbers: ");

    for (int i = 0; i < size; i++)

    {

        scanf("%d",(ptr+i));

    }

    for (int i = 0; i < size; i++)

    {

        sum=sum+ \*(ptr+i);

    }

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

    free(ptr);

    return 0;

}

Output;

enter the size of the array.5

enter the numbers: 45 65 78 21 91

the sum of the entered number is 300

PS C:\Users\tusha\Documents\coadind>

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

Input:

#include<iostream>

int main()

{

    int \*ptr, \*max,size;

    printf("enter the size :\n");

    scanf("%d",&size);

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

    printf("enter the element");

    for (int i = 0; i < size; i++)

    {

        scanf("%d",ptr+i);

    }

    for (int i = 0; i < (size-1); i++)

    {

        for(int j=i;j<size;j++)

        {

            if(\*(ptr+i)>\*(ptr+j))

            {

                max=ptr+i;

            }

            else

                max=ptr+j;

        }

    }

    printf("maximum number is %d",\*(max));

    return 0;

}

Output:

enter the size :

5

enter the element 5 1 3 4 9

maximum number is 9

PS C:\Users\tusha\Documents\coadind>

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

#include<iostream>

int main()

{

    int \*ptr;

    ptr=(int\*)malloc(sizeof(int));

    ptr=NULL;

    return 0;

}

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

Input:

#include<iostream>

int main()

{

    int \*ptr;

    ptr=(int\*)malloc(sizeof(int));

    \*ptr=10;

    printf("Before free : %d",\*ptr);

    free(ptr);

    printf("\nafter free : %d",\*ptr);

    return 0;

}

Ouput:

Before free : 10

after free : 16477976

PS C:\Users\tusha\Documents\coadind>

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

in C.

input;

#include<iostream>

void sort\_array(int \*, int );

void sort\_array(int \*p, int x)

{

    for (int i = 0; i < x-1; i++)

    {

        for(int j=i;j<x;j++)

        {

            if(\*(p+i)>\*(p+j))

            {

                int temp;

                temp=\*(p+i);

                \*(p+i)=\*(p+j);

                \*(p+j)=temp;

            }

        }

    }

}

int main()

{

    int \*ptr,size;

    printf("enter the size ");

    scanf("%d",&size);

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

    printf("enter the numbers: \n");

    for (int i = 0; i < size; i++)

    {

        scanf("%d",ptr+i);

    }

    sort\_array(ptr,size);

    printf("the sorted aray is : \n");

    for (int i = 0; i < size; i++)

    {

        printf("%d ", \*(ptr+i));

    }

    printf("\n  the maximum number is: %d ", \*(ptr+size-1));

    printf("\n the smallest number is : %d",\*ptr);

    return 0;

}

output:

enter the size 5

enter the numbers:

9 7 6 4 2

the sorted aray is :

2 4 6 7 9

the maximum number is: 9

the smallest number is : 2

PS C:\Users\tusha\Documents\coadind>