| **Name** | Jhaveri Varun Nimitt |
| --- | --- |
| **UID no.** | 2023800042 |
| **Experiment No.** | 4 |

| **AIM:** | Demonstrate the use of one-dimensional arrays to solve a given problem. |
| --- | --- |
| **Program 1** | |
| **PROBLEM STATEMENT :** | Write a C Program which contains a function to perform search of a particular element on an array. Create an array in main() and call the function to test it. |
| **ALGORITHM:** | 1. Start   * 2. Define a void function named display that takes an integer arrays[] and an integer size * 3. Start a for loop from i=0 to size-1, incrementing i by 1 each iteration   a. Print arrays[i] followed by a space   * 4. End for loop * 5. End function display * 6. Define an int function named ispresent that takes an integer array arrayforpresent[], an integer arraysize, and an integer whotocheck * 7. Declare an integer flag and initialize it to 0 * 8. Start a for loop from i=0 to arraysize-1, incrementing i by 1 each iteration   a. Check if arrayforpresent[i] is equal to whotocheck  i. If it is, set flag to 1 and return 1   * 9. If flag is 0, return 0 * 10. End function ispresent   11. Start main function  12. Declare an integer variable arrsize  13. Print "Enter size:"  14. Scan an integer into arrsize  15. Declare an integer array arr of size arrsize  16. Start a for loop from i=0 to arrsize-1, incrementing i by 1 each iteration  a. Print "Enter" (i+1) "element:"  b. Scan an integer into arr[i]  17. Call the display function with arguments arr and arrsize  18. Declare an integer variable checksum  19. Print "Enter value which u want to check for:"  20. Scan an integer into checksum  21. If ispresent(arr, arrsize, checksum) ==1  a. Print "yes,it is present"  22. Else  a. Print "no, it is not present"  23. End main function  24. End |
| **PROGRAM:** | #include <stdio.h> void display(int arrays[],int size); int ispresnt(int arrayforpresent[],int arraysize,int whotocheck); int main() {  int arrsize;  printf("Enter size :");  scanf("%d",&arrsize);  int arr[arrsize];  for(int i =0;i<arrsize;i++)  {  printf("Enter %d element:",(i+1));  scanf("%d",&arr[i]);  }  display(arr,arrsize);  int checksum;  printf("\nEnter value which u want to check for:");  scanf("%d",&checksum);  if (ispresnt(arr,arrsize,checksum)==1)  {  printf("\nyes,it is present");  }  else  {  printf("\n no, it is not present");  } }  void display(int arrays[],int size) {  for(int i=0;i<size;i++)  {  printf("%d ",arrays[i]);  } }  int ispresnt(int arrayforpresent[],int arraysize,int whotocheck) {  int flag =0;  for(int i=0;i<arraysize;i++)  {  if (arrayforpresent[i] == whotocheck)  {  flag=1;  return 1;  }    }  if(flag == 0)  {  return 0;  }   } |
| **RESULT:** | |
| **Program 2** | |
| **PROBLEM STATEMENT :** | Write a C Program which contains a function to sort an array using selection sort. Create an array in main() and call the function to test it. |
| **ALGORITHM:** | 1. Start   * 2. Define a void function named sort that takes an integer arrays[] and an integer size * 3. Declare integer variables i, j, and min\_index initializing them all to 0 * 4. Start a for loop from i=0 to size, incrementing i by 1 each iteration.   a. Set min\_index to i  b. Start a nested for loop from j=i+1 to size, incrementing j by 1 each iteration.  i. Check if arrays[j] is less than arrays[min\_index]  - If true, set min\_index to j  c. Declare an integer variable temp and initialize it to 0  d. Set temp to arrays[min\_index]  e. Swap arrays[min\_index] with arrays[i]  f. Swap arrays[i] with temp   * 5. Start a for loop from i=0 to size-1, incrementing i by 1 each iteration.   a. Print "New array :" followed by arrays[i] with a new line   * 6. End function sort   7. Start main function  8. Declare an integer variable arrsize  9. Print "Enter size:"  10. Scan an integer into arrsize  11. Declare an integer array arr of size arrsize  12. Start a for loop from i=0 to arrsize-1  a. Print "Enter" (i+1) "element:"  b. Scan an integer into arr[i]  13. Call the sort function with arguments arr and arrsize  14. End main function  15. End |
| **PROGRAM:** | #include <stdio.h> void sort(int arrays[],int size);  int main() {  int arrsize;  printf("Enter size :");  scanf("%d",&arrsize);  int arr[arrsize];  for(int i =0;i<arrsize;i++)  {  printf("Enter %d element:",(i+1));  scanf("%d",&arr[i]);  }  sort(arr,arrsize); }  void sort(int arrays[],int size) {  int i, j, min\_index=0;  for (i = 0; i < size; i++)   {  min\_index = i;  for (j = i+1; j < size; j++)  {  if (arrays[j] < arrays[min\_index])  {  min\_index = j;  }  }  int temp=0;   temp= arrays[min\_index];  arrays[min\_index] = arrays[i];  arrays[i] = temp;  }  for(int i =0;i<size;i++)  {  printf("\nNew array : %d",arrays[i]);   } } |
| **RESULT:** | |
| **CONCLUSION:** | **I have understood the way to use one-dimensional arrays to solve a given problem** |