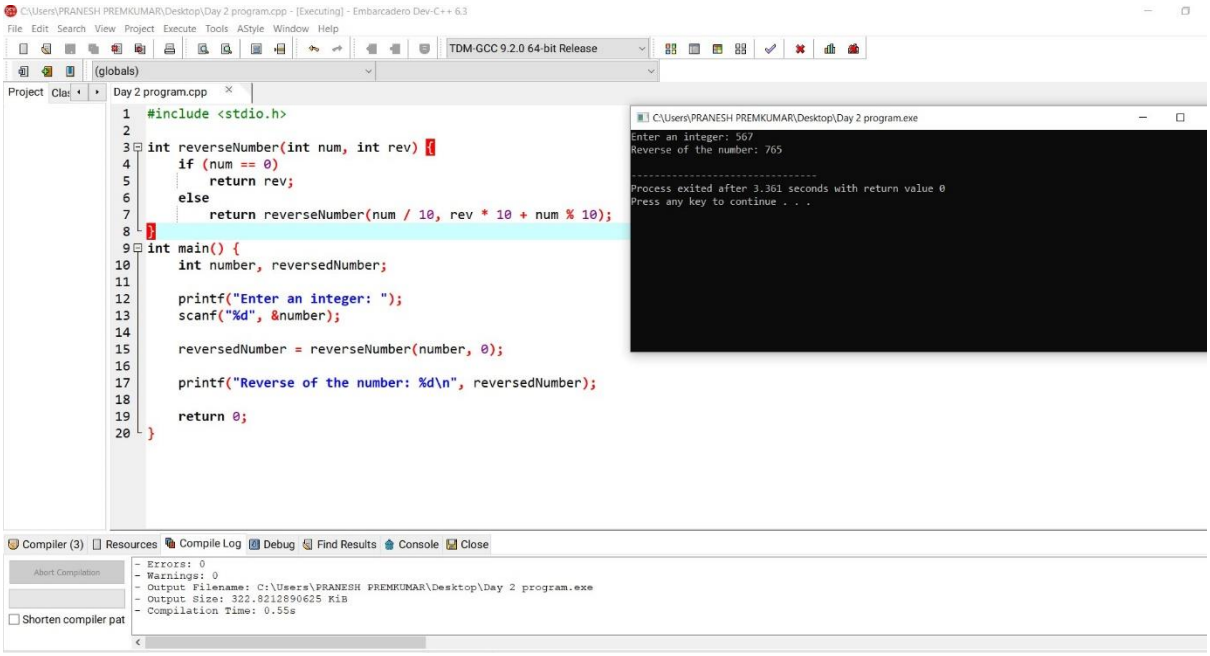
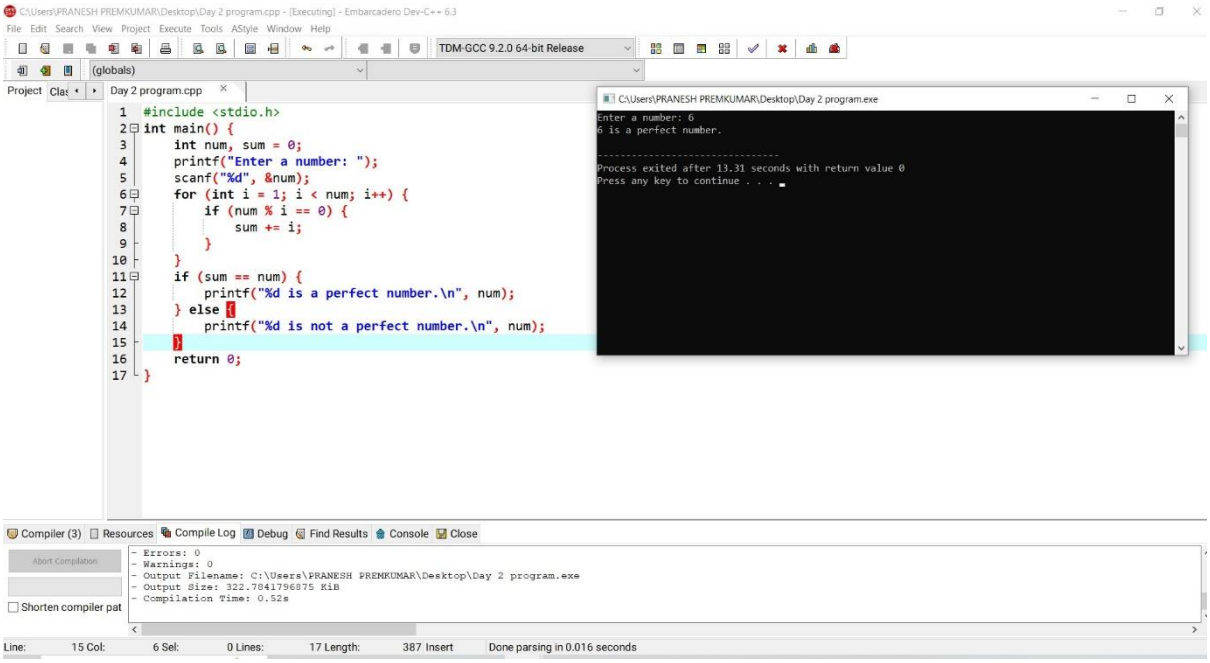
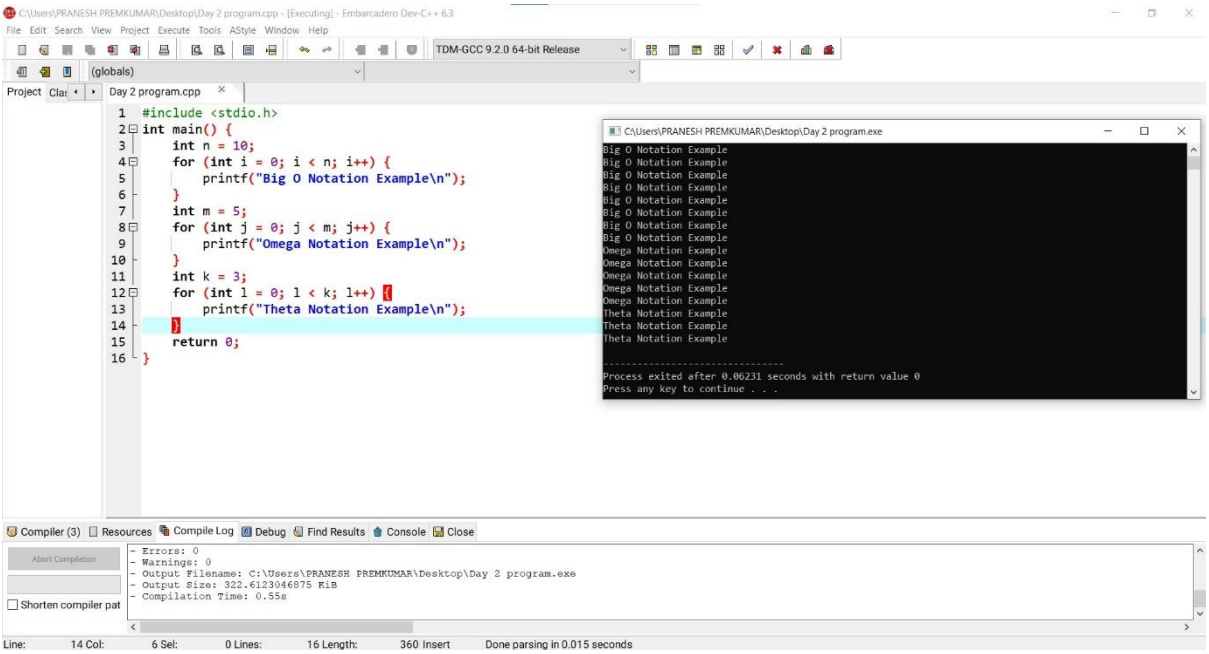
1.Reverse a number



2.Perfect number



3.

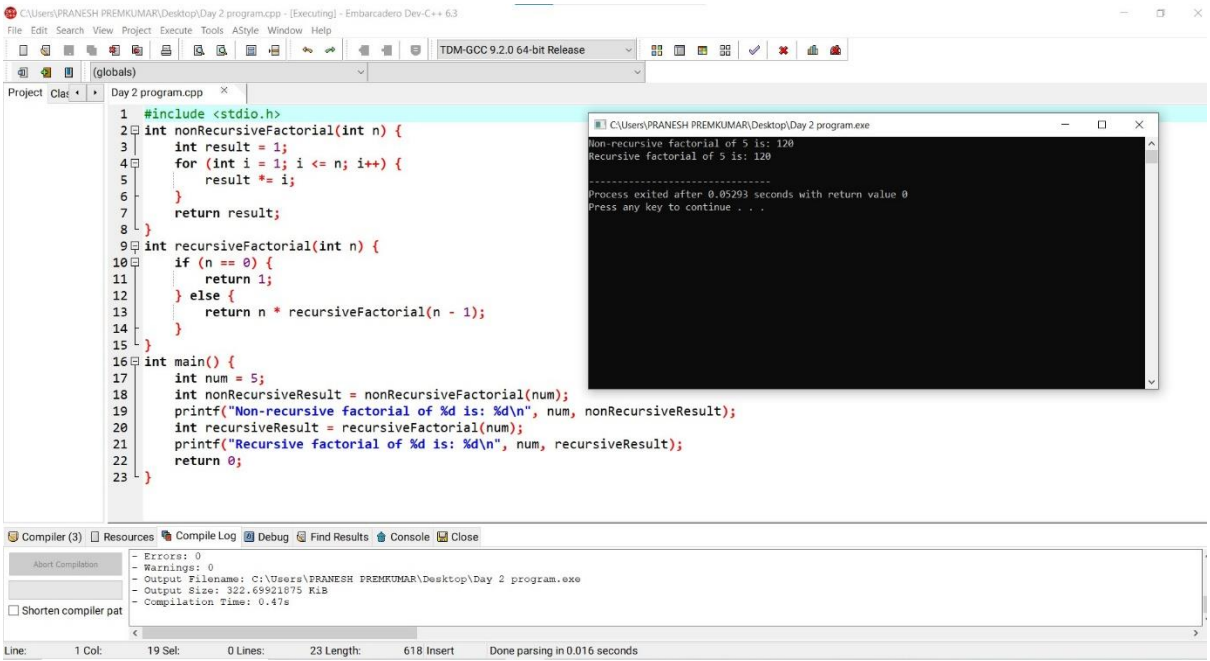


```c
#include <stdio.h>
int main() {
    int n = 10;
    for (int i = 0; i < n; i++) {
        printf("Big O Notation Example\n");
    }
    int m = 5;
    for (int j = 0; j < m; j++) {
        printf("Omega Notation Example\n");
    }
    int k = 3;
    for (int l = 0; l < k; l++) {
        printf("Theta Notation Example\n");
    }
    return 0;
}
```

```
Big O Notation Example
Big O Notation Example
Big O Notation Example
Big O Notation Example
Big O Notation Example
Big O Notation Example
Big O Notation Example
Big O Notation Example
Omega Notation Example
Omega Notation Example
Omega Notation Example
Omega Notation Example
Omega Notation Example
Theta Notation Example
Theta Notation Example
Theta Notation Example

--------------------------------
Process exited after 0.06231 seconds with return value 0
Press any key to continue . . .
```

4.

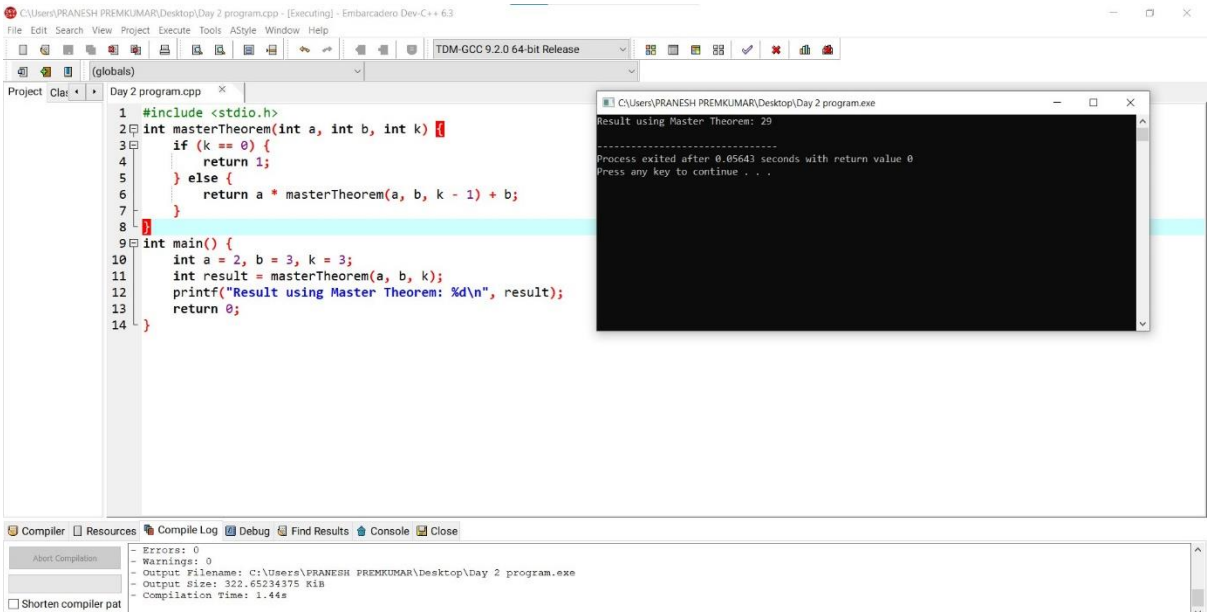


```c
#include <stdio.h>
int nonRecursiveFactorial(int n) {
    int result = 1;
    for (int i = 1; i <= n; i++) {
        result *= i;
    }
    return result;
}
int recursiveFactorial(int n) {
    if (n == 0) {
        return 1;
    } else {
        return n * recursiveFactorial(n - 1);
    }
}
int main() {
    int num = 5;
    int nonRecursiveResult = nonRecursiveFactorial(num);
    printf("Non-recursive factorial of %d is: %d\n", num, nonRecursiveResult);
    int recursiveResult = recursiveFactorial(num);
    printf("Recursive factorial of %d is: %d\n", num, recursiveResult);
    return 0;
}
```

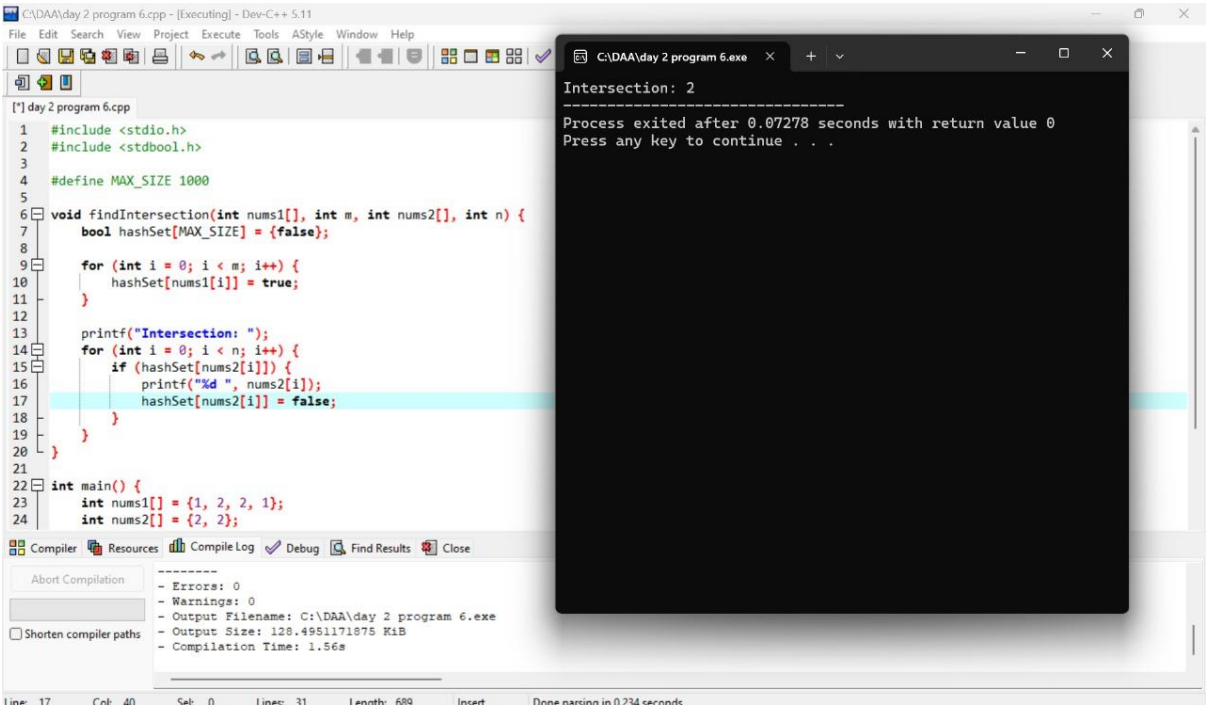
```
Non-recursive factorial of 5 is: 120
Recursive factorial of 5 is: 120

--------------------------------
Process exited after 0.05293 seconds with return value 0
Press any key to continue . . .
```

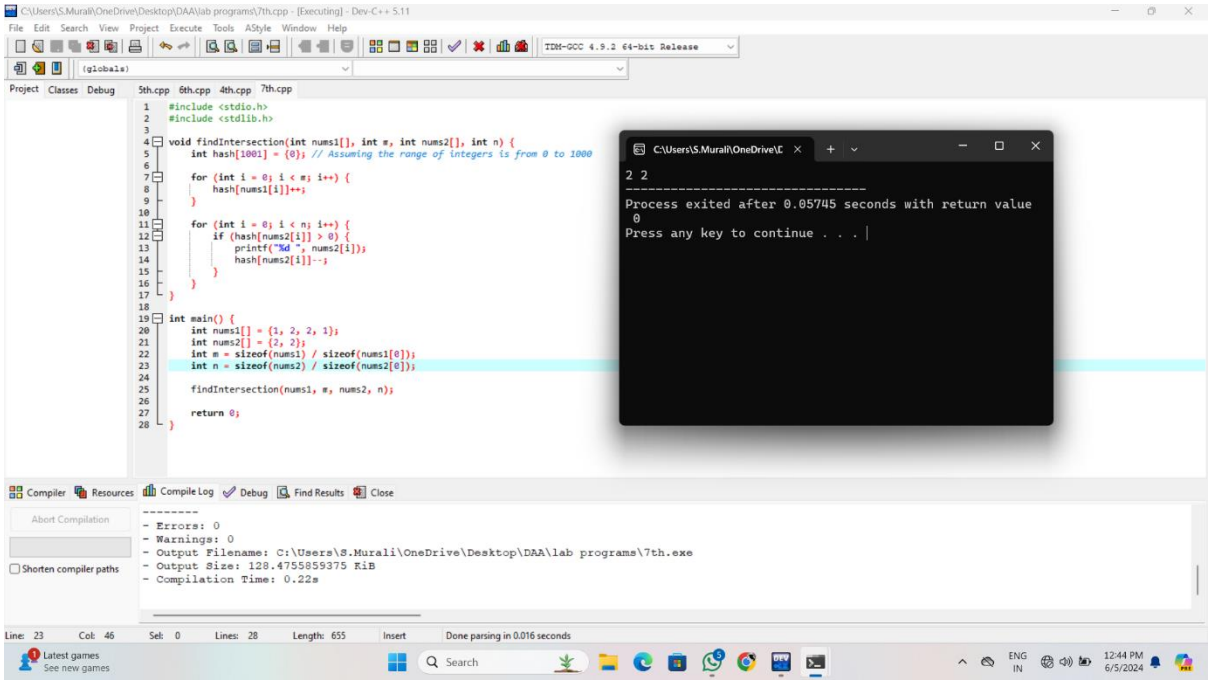
5:Master theorem


C:\Users\PRANESH PREMKUMAR\Desktop\Day 2 program.cpp - [Executing] - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
Project
Day 2 program.cpp
#include <stdio.h>
int masterTheorem(int a, int b, int k) {
if (k == 0) {
return 1;
} else {
return a * masterTheorem(a, b, k - 1) + b;
}
}
int main() {
int a = 2, b = 3, k = 3;
int result = masterTheorem(a, b, k);
printf("Result using Master Theorem: %d\n", result);
return 0;
}
C:\Users\PRANESH PREMKUMAR\Desktop\Day 2 program.exe
Result using Master Theorem: 29
Process exited after 0.05643 seconds with return value 0
Press any key to continue . . .
Compiler
Resources
Compile Log
Debug
Find Results
Console
Close
Abort Compilation
Shorten compiler pat
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\PRANESH PREMKUMAR\Desktop\Day 2 program.exe
- Output Size: 322.65234375 KiB
- Compilation Time: 1.44s

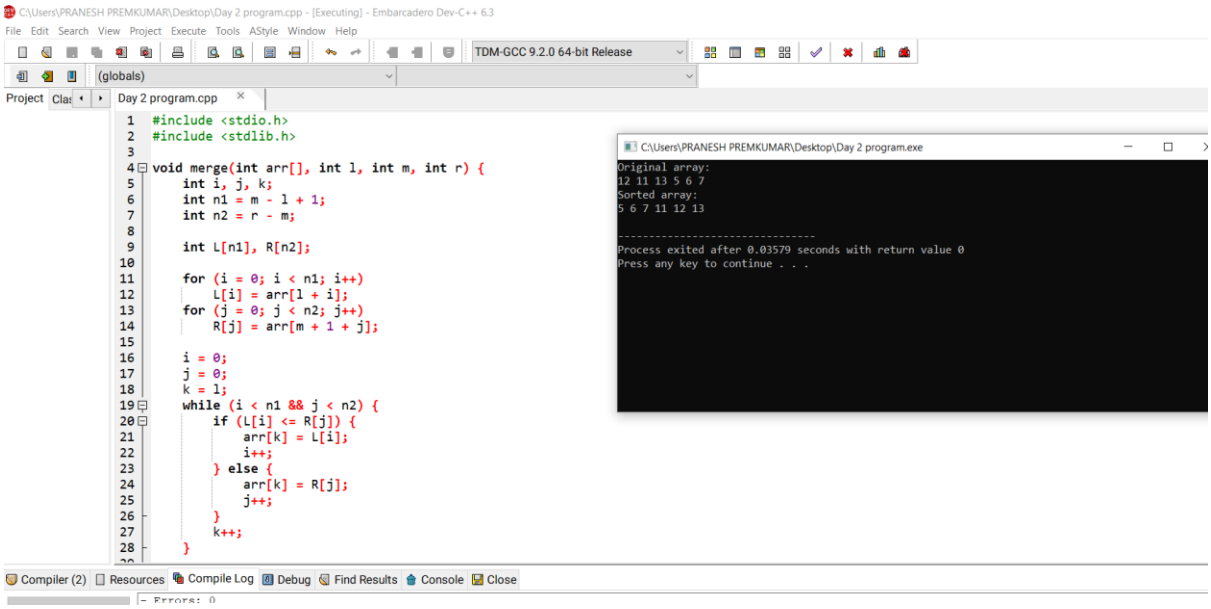
6.


C:\DAA\day 2 program 6.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
[*] day 2 program 6.cpp
#include <stdio.h>
#include <stdbool.h>
#define MAX_SIZE 1000
void findIntersection(int nums1[], int m, int nums2[], int n) {
bool hashSet[MAX_SIZE] = {false};
for (int i = 0; i < m; i++) {
hashSet[nums1[i]] = true;
}
printf("Intersection: ");
for (int i = 0; i < n; i++) {
if (hashSet[nums2[i]]) {
printf("%d ", nums2[i]);
hashSet[nums2[i]] = false;
}
}
}
int main() {
int nums1[] = {1, 2, 2, 1};
int nums2[] = {2, 2};
C:\DAA\day 2 program 6.exe
Intersection: 2
Process exited after 0.07278 seconds with return value 0
Press any key to continue . . .
Compiler
Resources
Compile Log
Debug
Find Results
Close
Abort Compilation
Shorten compiler paths
- Errors: 0
- Warnings: 0
- Output Filename: C:\DAA\day 2 program 6.exe
- Output Size: 128.4951171875 KiB
- Compilation Time: 1.56s

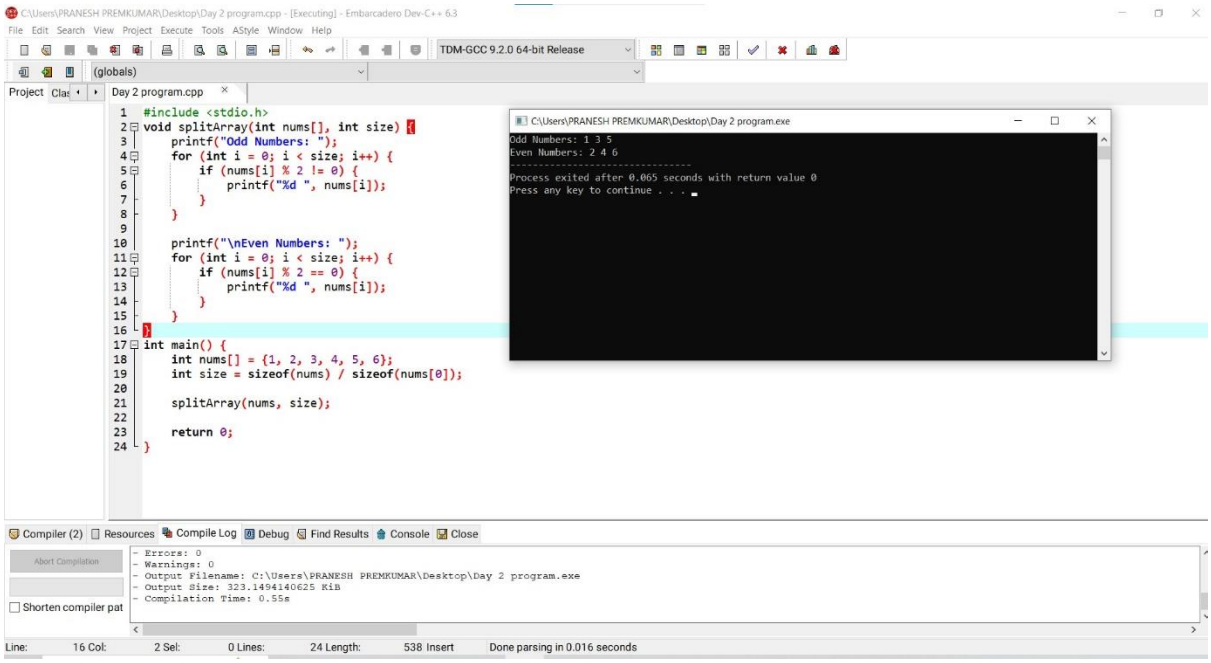
7.


C:\Users\S.Murali\OneDrive\Desktop\DAA\lab programs\7th.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
5th.cpp 6th.cpp 4th.cpp 7th.cpp
#include <stdio.h>
#include <stdlib.h>
void findIntersection(int nums1[], int m, int nums2[], int n) {
int hash[1001] = {0}; // Assuming the range of integers is from 0 to 1000
for (int i = 0; i < m; i++) {
hash[nums1[i]]++;
}
for (int i = 0; i < n; i++) {
if (hash[nums2[i]] > 0) {
printf("%d ", nums2[i]);
hash[nums2[i]]--;
}
}
}
int main() {
int nums1[] = {1, 2, 2, 1};
int nums2[] = {2, 2};
int m = sizeof(nums1) / sizeof(nums2[0]);
int n = sizeof(nums2) / sizeof(nums2[0]);
findIntersection(nums1, m, nums2, n);
return 0;
}
C:\Users\S.Murali\OneDrive\D
2 2
Process exited after 0.05745 seconds with return value 0
Press any key to continue . . .
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\S.Murali\OneDrive\Desktop\DAA\lab programs\7th.exe
- Output Size: 128.4755859375 KiB
- Compilation Time: 0.22s
Line: 23 Col: 46 Sel: 0 Lines: 28 Length: 655 Insert Done parsing in 0.016 seconds
Latest games
See new games
Search
ENG
IN
12:44 PM
6/5/2024

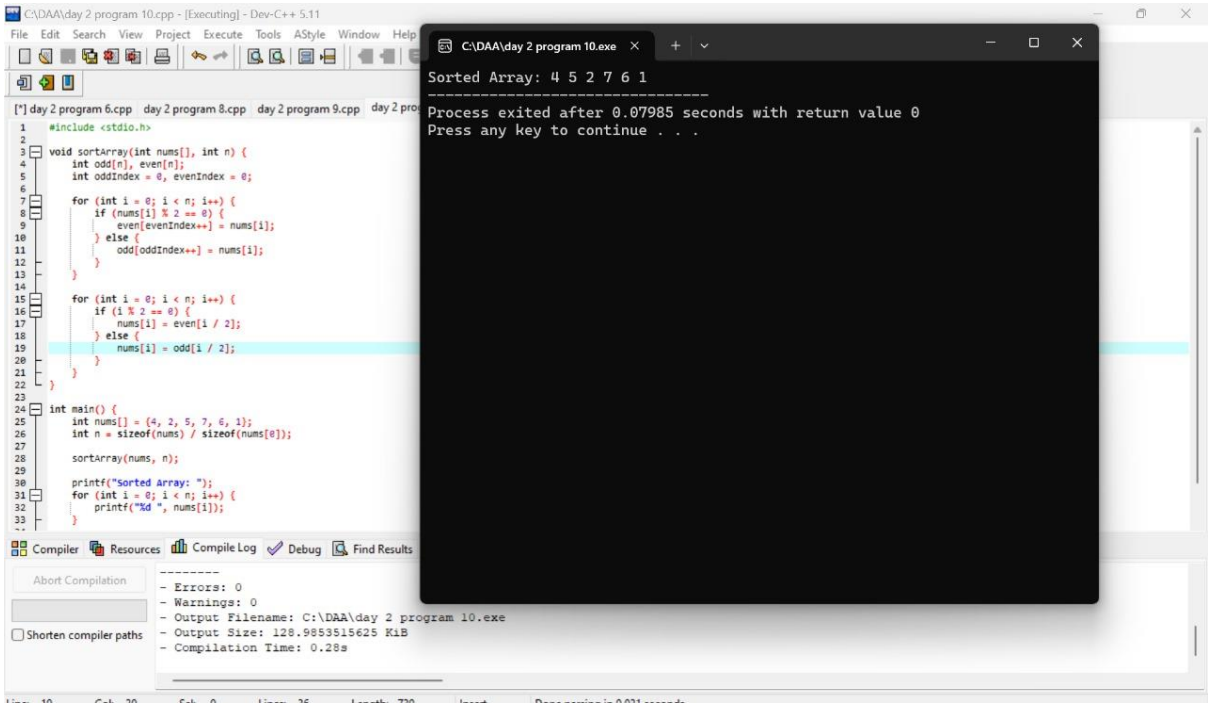
8.


C:\Users\PRANESH PREMKUMAR\Desktop\Day 2 program.cpp - [Executing] - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
Project
Day 2 program.cpp
#include <stdio.h>
#include <stdlib.h>
void merge(int arr[], int l, int m, int r) {
int i, j, k;
int n1 = m - l + 1;
int n2 = r - m;
int L[n1], R[n2];
for (i = 0; i < n1; i++)
L[i] = arr[l + i];
for (j = 0; j < n2; j++)
R[j] = arr[m + 1 + j];
i = 0;
j = 0;
k = l;
while (i < n1 && j < n2) {
if (L[i] <= R[j]) {
arr[k] = L[i];
i++;
} else {
arr[k] = R[j];
j++;
}
k++;
}
C:\Users\PRANESH PREMKUMAR\Desktop\Day 2 program.exe
Original array:
12 11 13 5 6 7
Sorted array:
5 6 7 11 12 13
Process exited after 0.03579 seconds with return value 0
Press any key to continue . . .
Compiler (2) Resources Compile Log Debug Find Results Console Close
- Errors: 0

9.


C:\Users\PRANESH PREMKUMAR\Desktop\Day 2 program.cpp - [Executing] - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
Project
Day 2 program.cpp
#include <stdio.h>
void splitArray(int nums[], int size) {
printf("Odd Numbers: ");
for (int i = 0; i < size; i++) {
if (nums[i] % 2 != 0) {
printf("%d ", nums[i]);
}
}
printf("\nEven Numbers: ");
for (int i = 0; i < size; i++) {
if (nums[i] % 2 == 0) {
printf("%d ", nums[i]);
}
}
}
int main() {
int nums[] = {1, 2, 3, 4, 5, 6};
int size = sizeof(nums) / sizeof(nums[0]);
splitArray(nums, size);
return 0;
}
C:\Users\PRANESH PREMKUMAR\Desktop\Day 2 program.exe
Odd Numbers: 1 3 5
Even Numbers: 2 4 6
Process exited after 0.065 seconds with return value 0
Press any key to continue . . .
Compiler (2)
Resources
Compile Log
Debug
Find Results
Console
Close
Abort Compilation
Shorten compiler pat
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\PRANESH PREMKUMAR\Desktop\Day 2 program.exe
- Output Size: 323.1494140625 KiB
- Compilation Time: 0.55s
Line: 16 Col: 2 Sel: 0 Lines: 24 Length: 538 Insert
Done parsing in 0.016 seconds

10.


C:\DAA\day 2 program 10.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
[*] day 2 program 6.cpp
day 2 program 8.cpp
day 2 program 9.cpp
#include <stdio.h>
void sortArray(int nums[], int n) {
int odd[n], even[n];
int oddIndex = 0, evenIndex = 0;
for (int i = 0; i < n; i++) {
if (nums[i] % 2 == 0) {
even[evenIndex++] = nums[i];
} else {
odd[oddIndex++] = nums[i];
}
}
for (int i = 0; i < n; i++) {
if (i % 2 == 0) {
nums[i] = even[i / 2];
} else {
nums[i] = odd[i / 2];
}
}
}
int main() {
int nums[] = {4, 2, 5, 7, 6, 1};
int n = sizeof(nums) / sizeof(nums[0]);
sortArray(nums, n);
printf("Sorted Array: ");
for (int i = 0; i < n; i++) {
printf("%d ", nums[i]);
}
C:\DAA\day 2 program 10.exe
Sorted Array: 4 5 2 7 6 1
Process exited after 0.07985 seconds with return value 0
Press any key to continue . . .
Compiler
Resources
Compile Log
Debug
Find Results
Abort Compilation
Shorten compiler paths
- Errors: 0
- Warnings: 0
- Output Filename: C:\DAA\day 2 program 10.exe
- Output Size: 128.9853515625 KiB
- Compilation Time: 0.28s