B.BHANUTEJA REDDY-192325016

21.Develop a C program to implement the worst fit algorithm of memory management.

AIM:

To develop a C program that implements the **Worst Fit** memory management algorithm for allocating processes to memory blocks.

ALGORITHM:

1. **Start.**
2. Input the number of memory blocks and their sizes.
3. Input the number of processes and their sizes.
4. For each process:
    - Find the memory block with the largest size that can accommodate the process.
    - Allocate the process to this block and update the block size by subtracting the process size.
5. If no suitable block is found, the process remains unallocated.
6. Display the allocation result.
7. **End.**

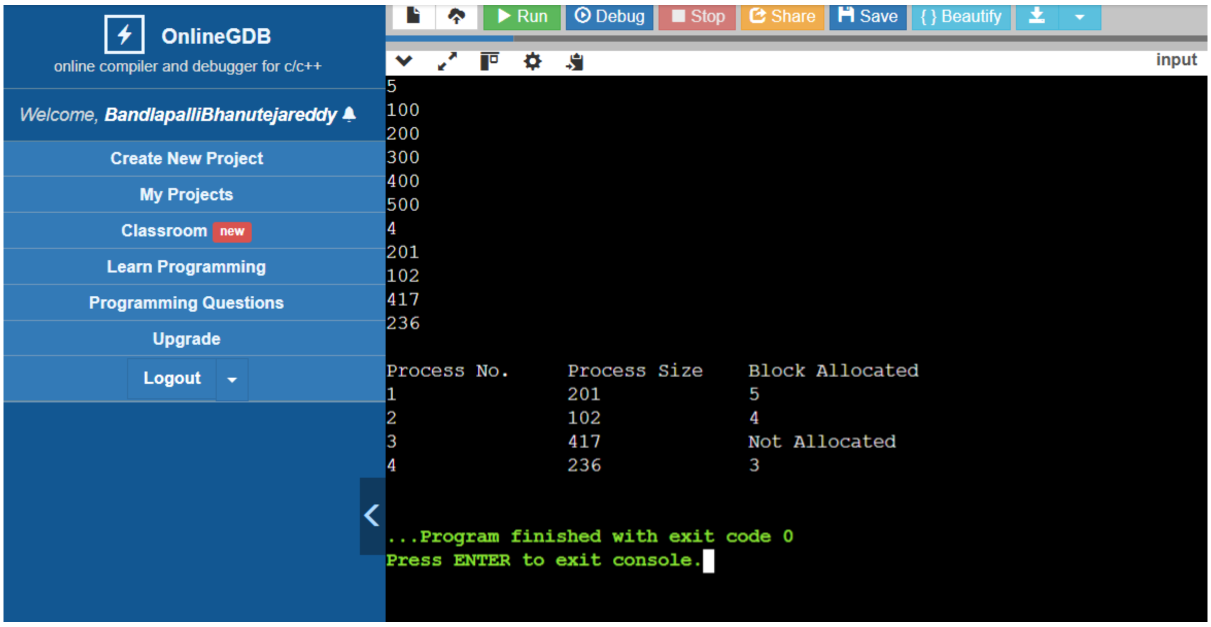
PROCEDURE:

1. **Define arrays** to store memory block sizes and process sizes.
2. Use loops to iterate through processes and blocks.
3. Compare block sizes to find the largest block that can accommodate the current process.
4. Update the block size after allocation.
5. Print the allocation table showing which block is assigned to each process.

```
#include <stdio.h>
```

```
int main() {
  int nb, np;
  scanf("%d", &nb);
  int blockSize[nb];
  for (int i = 0; i < nb; i++) {
    scanf("%d", &blockSize[i]);
  }
  scanf("%d", &np);
  int processSize[np], allocation[np];
  for (int i = 0; i < np; i++) {
    scanf("%d", &processSize[i]);
    allocation[i] = -1;
  }
  for (int i = 0; i < np; i++) {
    int worstIdx = -1;
    for (int j = 0; j < nb; j++) {
      if (blockSize[j] >= processSize[i]) {
        if (worstIdx == -1 || blockSize[j] > blockSize[worstIdx]) {
          worstIdx = j;
        }
      }
    }
    if (worstIdx != -1) {
      allocation[i] = worstIdx;
      blockSize[worstIdx] -= processSize[i];
    }
  }
  printf("\nProcess No.\tProcess Size\tBlock Allocated\n");
  for (int i = 0; i < np; i++) {
    printf("%d\t\t%d\t\t", i + 1, processSize[i]);
    if (allocation[i] != -1)
      printf("%d\n", allocation[i] + 1);
    else
      printf("Not Allocated\n");
  }
  return 0;
}
```

OUTPUT:



```
5
100
200
300
400
500
4
201
102
417
236

Process No.     Process Size    Block Allocated
1               201             5
2               102             4
3               417             Not Allocated
4               236             3


...Program finished with exit code 0
Press ENTER to exit console.
```