#include <stdio.h>

#define MAX\_BLOCKS 10

#define MAX\_PROCESSES 5

void bestFit(int blocks[], int block\_size, int processes[], int process\_size) {

int allocation[MAX\_PROCESSES];

for (int i = 0; i < process\_size; i++) allocation[i] = -1;

for (int i = 0; i < process\_size; i++) {

int best\_idx = -1;

for (int j = 0; j < block\_size; j++) {

if (blocks[j] >= processes[i] && (best\_idx == -1 || blocks[j] < blocks[best\_idx])) {

best\_idx = j;

}

}

if (best\_idx != -1) {

allocation[i] = best\_idx;

blocks[best\_idx] -= processes[i];

}

}

printf("\nProcess No\tProcess Size\tBlock No\tBlock Size\n");

for (int i = 0; i < process\_size; i++) {

if (allocation[i] != -1) printf("%d\t\t%d\t\t%d\t\t%d\n", i + 1, processes[i], allocation[i] + 1, blocks[allocation[i]]);

else printf("%d\t\t%d\t\tNot Allocated\n", i + 1, processes[i]);

}

}

int main() {

int blocks[MAX\_BLOCKS] = {50, 200, 100, 500}, processes[MAX\_PROCESSES] = {120, 50, 200};

bestFit(blocks, 4, processes, 3);

return 0;

}

