Q2: Part 1

```
B-F( bArray){
    n = bArray.size()
    element = [0,1], final_restult = Null;
    for (i from 1 to 2^n){                          <- (2^n)
        res = list()
        for (j from 1 to n){
            If element[i % 2] == 1 {
                res.append(bArray[j])
            }
        }
        valid = 0
        K = res.size()
        for (c from 1 to K-1){
            If (res[c+1] - res[c]) > C {
                valid = 1
                break
            }
        }
        if (final_result == Null and valid == 0){
            final_result = res;
        }
        eles{
            If (final_resht.size() >= res.size() and valid == 0){
                final_result = res
            }
        }.
} }
```

Generate one possible result

Check validation

Assign the result.

☆ ANALYsis: ☆

The run-time of this code is $O(2^n \cdot n)$.

Scanned with CamScanner

Part 2.

```
Greedy ( bArray){
    n = bArray.size()
    A = {b[n-1]}
    X = n-1
    while ( X ≠ 0) (
        Let P be the longest break point, bArray [P] + C ≤ bArray[X]
        X = P
        A = A + [bArray[P]]
    }
}.
```

Q3: sub-optimality:

If $A = \{O_1, O_2, \cdots O_k\}$ is an optimal solution, than $A' = A - \{O_1\}$ is the optimal solution of $B' = \{b_1, b_2, \cdots b_p \mid \text{where } b_p \text{ is the largest break point}, b_p + C \leq b_n\}$.

Proof: If there exist an optimal solution O' for B' that $|O'| < |A'|$, then, Let $O = O' + \{O_1\}$ which means O is the globle optimal solution and

$$|O| < |A|$$

this means A is not an globle optimal solution. which contradict with the given information. Out Assumption is not right. So, the A' is the optimal solution for B'.