Algorithm 4.4 Scheduling with Deadlines

Problem: Determine the schedule with maximum total profit given that each job has a profit that will be obtained only if the job is scheduled by its deadline

Inputs:

n - the number of jobs

deadline - array with deadline values, indexed 1 to n, where deadline[i] is the deadline for the ith job; array has been sorted in nonincreasing order based on profit

job - array with job number values that correspond to sorted deadline[]

profits - array with profit values that correspond to sorted deadline[]

Output: returns optimal sequence J for the jobs

```
int J[] schedule(int n, int deadline[])

    J = 1  // array J has index 1

    loop for all jobs in deadline[]  i=2 to n
        K = J with next ith element index   // copy elements over from J to K
                                            // and add next element's index
        if (K is feasible)
            J = K  // copy contents of K array to J array
    return J

boolean feasible(K)
    tmp[] = K // copy values of K to array tmp
    boolean isFeasible = true

    loop for all elements in tmp  (index i=0 to size of tmp)
        loop for all elements start with second element in tmp (index j=1 to size of tmp)
            index1 = tmp[i]
            index2 = tmp[j]
            if (deadline[index1] > deadline[index2])
                swap job numbers in tmp

    loop for all jobs in tmp k=0 to size of tmp
        job = tmp[k]
        if (deadline[job] < k+1)
            isFeasible = false
            break

    return isFeasible
```

sorted jobs:

```
deadline        0 1 1 2 2
job             0 4 2 1 3
profit          0 40 35 30 25
n = 4

J = 1  // first job in sorted array    job 4

i = 2; K = 1 2   // same as J plus next job: job[1] and job[2] which is [4 2]
        feasible?  tmp = 1 2
        isFeasible=true
        loop i =0
                j = 1 :   index1=tmp[0] = 1    index2=tmp[1]=1  //deadline values
                          if 1 > 1 false
        loop    k=0       job=tmp[0]=1  if 1 < 1 false
                k=1       job=tmp[1]=1  if 1 < 2 true   : isFeasible = False
        return False

i=3; K = 1 3   2   // same as J plus next job: job[1] and job[3] which is [4 1]
        feasible?  tmp = 1 3
        isFeasible=true
        loop i =0
                j = 1 :   index1=tmp[0] = 1    index2=tmp[2]=2  //deadline values
                          if 1 > 2 false
        loop    k=0       job=tmp[0]=1  if 1 < 1 false
                k=1       job=tmp[3]=2  if 2 < 2  false
        return True

        J = 1 3         // jobs 4 1

i=4; K = 1 3 4
        feasible?  tmp = 1 3 4
        isFeasible=true
        loop i =0
                j = 1 :   index1=tmp[0] = 1    index2=tmp[2]=2  //deadline values
                          if 1 > 2 false
                j=2 :     index1=tmp[0]=1       index2=tmp[3]=2
                          if 1 > 2 false
             i=1
                j=2 :     index1=tmp[2] = 2    index2=tmp[3]=2  //deadline values
                          if 2 > 2 false

        loop    k=0       job=tmp[0]=1  if 1 < 1 false
                k=1       job=tmp[3]=2  if 2 < 2  false
                k=2       job=tmp[4]=2  if 2 < 3  true :  isFeasible = False
        return False
```

J = 1 3   // job[1]=4   job[3]=1

So final set 4 1; feasible sequence 4 1; and profit 70