Name :- Ashish Verma Soni

Reg no:- RA1911042010102

Section:- CSBS Q2

**Aim****: Implementation of Job scheduling problem**

**Algorithm:**

Algorithm JOB\_SCHEDULING( J, D, P )

// Description : Schedule the jobs using the greedy approach which maximizes the profit

// Input :

J: Array of N jobs

D: Array of the deadline for each job

P: Array of profit associated with each job

// Output : Set of scheduled job which gives maximum profit

Sort all jobs in J in decreasing order of profit

S ← Φ // S is set of scheduled jobs, initially it is empty

SP ← 0 // Sum is the profit earned

for i ← 1 to N do

if Job J[i] is feasible then

Schedule the job in the latest possible free slot meeting its deadline.

S ← S ∪ J[i]

SP ← SP + P[i]

end

end

**Source Code:**

**PROGRAM:**

def printJobScheduling(arr, t):

n = len(arr)

for i in range(n):

for j in range(n - 1 - i):

if arr[j][2] < arr[j + 1][2]:

arr[j], arr[j + 1] = arr[j + 1], arr[j]

result = [False] \* t

job = ['-1'] \* t

for i in range(len(arr)):

for j in range(min(t - 1, arr[i][1] - 1), -1, -1):

if result[j] is False:

result[j] = True

job[j] = arr[i][0]

break

print(job)

# Driver Code

arr = [['a', 2, 100], # Job Array

['b', 1, 19],

['c', 2, 27],

['d', 1, 25],

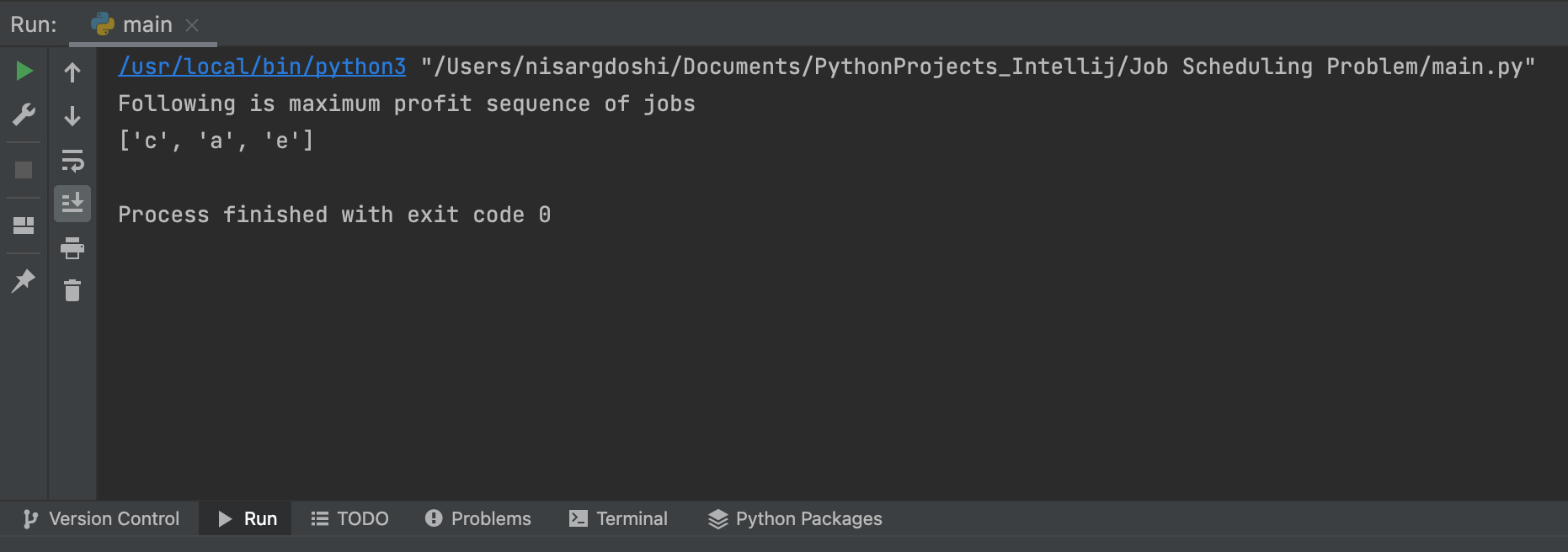
['e', 3, 15]]

print("Following is maximum profit sequence of jobs")

# Function Call

printJobScheduling(arr, 3)

**OUTPUT:**



**Result:** We have successfully implemented Implementation of Job scheduling problem.