**Прізвище:** Колісник

**Ім’я:** Андрій

**Група:** КН-405

**Кафедра.:** Кафедра Систем

Автоматизованого Проектування

**Дисципліна:** Теорія прийняття рішень

**Перевірив:** Кривий Р.З.

**Звіт**

До лабораторної роботи №2

На тему “Моделі прийняття рішень. Дерево рішень (Використання дерева рішень на прикладі рішення про будівництва заводу)”

**Мета роботи:** Одержання практичних навичок використання дерева рішень для рішення проблем.

**Короткі теоретичні відомості**

**Дерево рішень** (також можуть називатися деревами класифікацій або регресійними деревами) — використовується в галузі статистики та аналізу даних для прогнозних моделей. Структура дерева містить такі елементи: «листя» і «гілки». На ребрах («гілках») дерева ухвалення рішення записані атрибути, від яких залежить цільова функція, в «листі» записані значення цільової функції, а в інших вузлах — атрибути, за якими розрізняються випадки. Щоб класифікувати новий випадок, треба спуститися по дереву до листа і видати відповідне значення. Подібні дерева рішень широко використовуються в інтелектуальному аналізі даних. Мета полягає в тому, щоб створити модель, яка прогнозує значення цільової змінної на основі декількох змінних на вході.

**Індивідуальне завдання:**

Задача.

Компанія розглядає питання про будівництво заводу. Можливі три варіанти:

А) Побудувати великий завод вартістю М1 тис. доларів. При цьому варіанті можливі великий попит (річний дохід в розмірі D1 тис. доларів протягом наступних 5 років) з ймовірністю Р1 і низький попит (щорічні збитки D2 тис. доларів) з ймовірністю Р2 .

Б) Побудувати маленький завод вартістю М2 тис. Доларів. При цьому варіанті можливі великий попит (річний дохід в розмірі D1 тис. Доларів протягом наступних 5 років) з ймовірністю Р1 і низький попит (щорічні збитки D2 тис. доларів) з ймовірністю Р2

В) Відкласти будівництво заводу на 1 рік для збору додаткової інформації, яка може бути позитивною або негативною з ймовірністю Р3 і Р4 відповідно. У разі позитивної інформації можна побудувати заводи з зазначеним вище розцінками, а ймовірності великого і низького попиту змінюються на Р1 і Р2 відповідно. Доходи на наступні 4 роки залишаються колишніми. У разі негативної інформації компанія заводи будувати не буде.

**Порядок вирішення завдання:**

1) Зобразити дерево рішень, що відповідає умовам завдання.

2) Провести розрахунок очікуваних доходів для всіх вузлів.

3) Вибрати найбільш ефективний варіант рішення.

4) Описати порядок виконання роботи.

5) Реалізувати програмне забезпечення, яке б розв'язувало дану задачу. Мова програмування неважлива. Обов'язково: дані мають зчитуватись з файлу і виводитись у табличній формі.

Варіант № 13

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Вар. |  |  | A |  |  |  |  | Б |  |  |  |  | B |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | M1 | D1 | P1 | D2 | P2 | M2 | D1 | P1 | D2 | P2 | P3 | P4 | P1 | P2 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 13 | 900 | 250 | 0.8 | -60 | 0.2 | 300 | 190 | 0.8 | -45 | 0.2 | 0.8 | 0.2 | 0.9 | 0.1 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Дані для даної задачі зчитуються з зовнішнього файлу (Рис. 1)

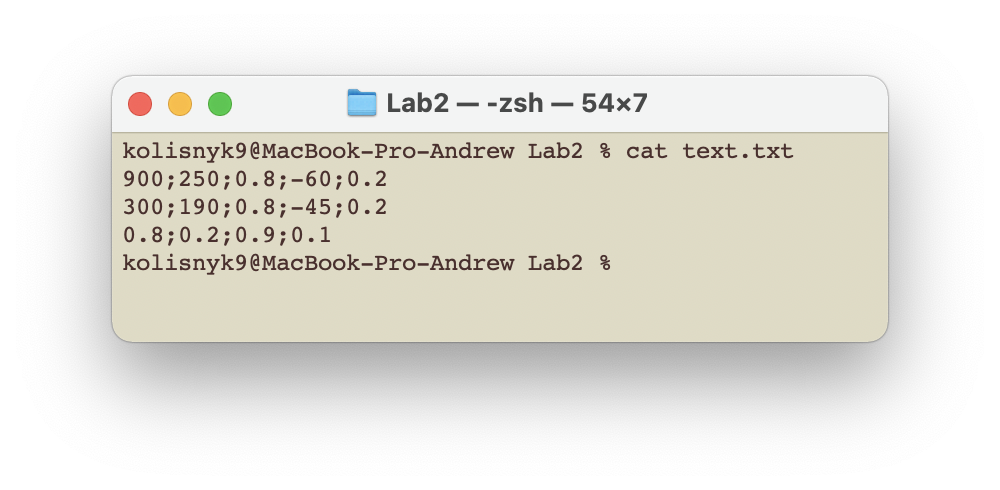


Рис. 1 Файл text.txt

**A)** Побудувати великий завод вартістю М1 тис. доларів. При цьому варіанті можливі великий попит (річний дохід в розмірі D1 тис. доларів протягом наступних 5 років) з ймовірністю Р1 і низький попит (щорічні збитки D2 тис. доларів) з ймовірністю Р2 .

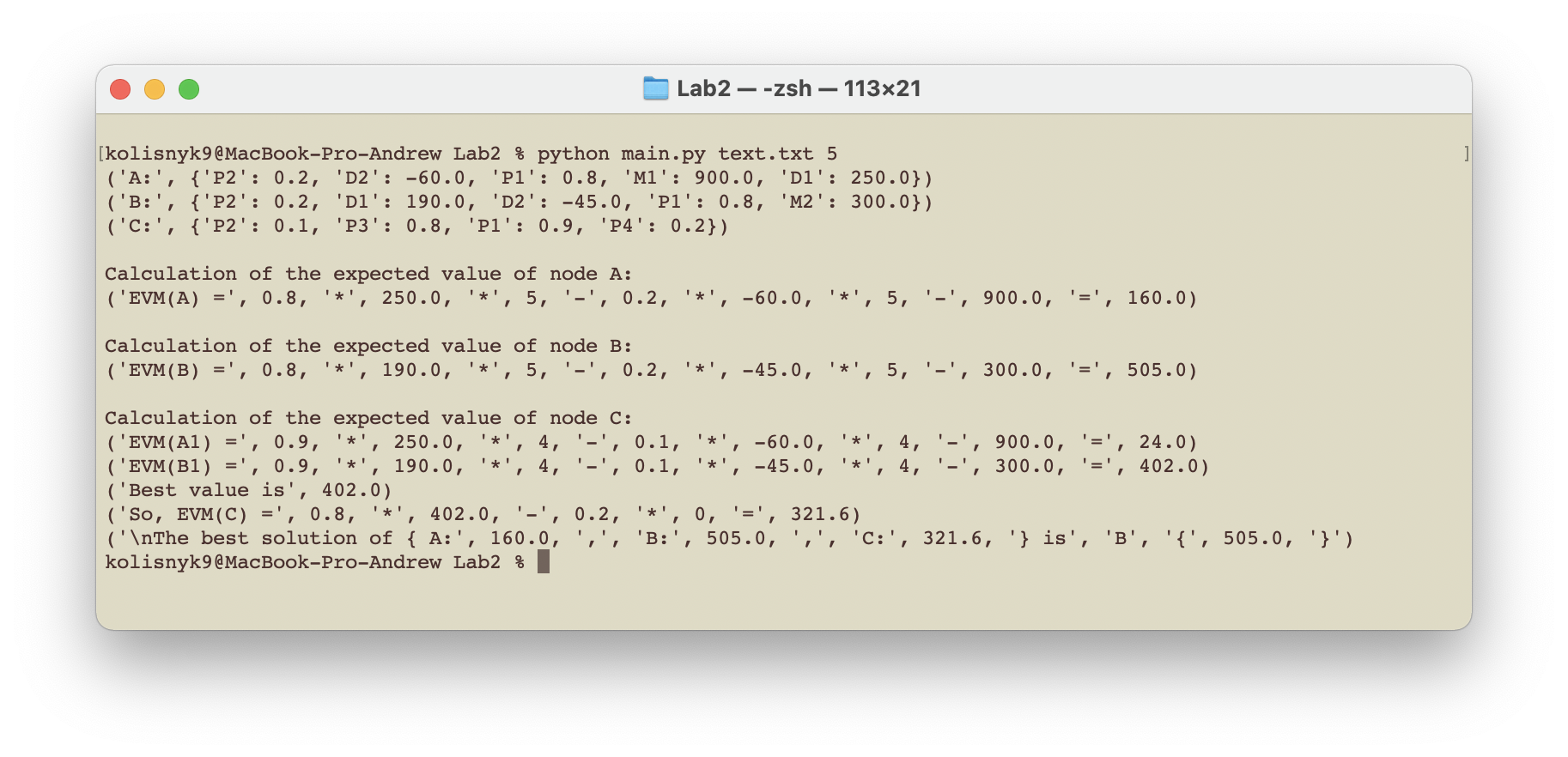


Рис. 2 Побудова великого заводу вартістю M1 тис. доларів

**Б)** Побудувати маленький завод вартістю М2 тис. Доларів. При цьому варіанті можливі великий попит (річний дохід в розмірі D1 тис. Доларів протягом наступних 5 років) з ймовірністю Р1 і низький попит (щорічні збитки D2 тис. доларів) з ймовірністю Р2

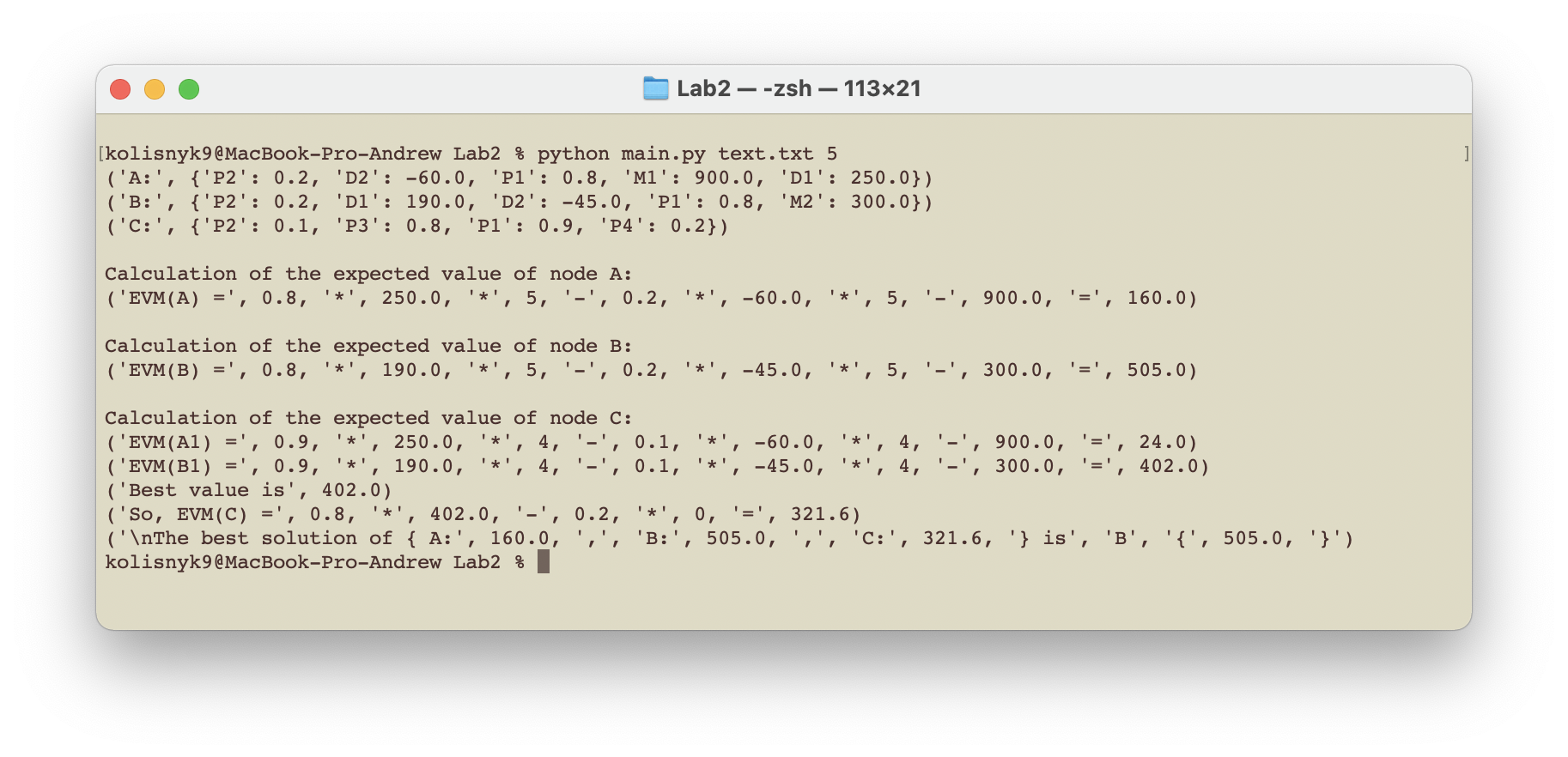


Рис. 3 Побудова великого заводу вартістю M2 тис. доларів

**В)** Відкласти будівництво заводу на 1 рік для збору додаткової інформації, яка може бути позитивною або негативною з ймовірністю Р3 і Р4 відповідно. У разі позитивної інформації можна побудувати заводи з зазначеним вище розцінками, а ймовірності великого і низького попиту змінюються на Р1 і Р2 відповідно. Доходи на наступні 4 роки залишаються колишніми. У разі негативної інформації компанія заводи будувати не буде.

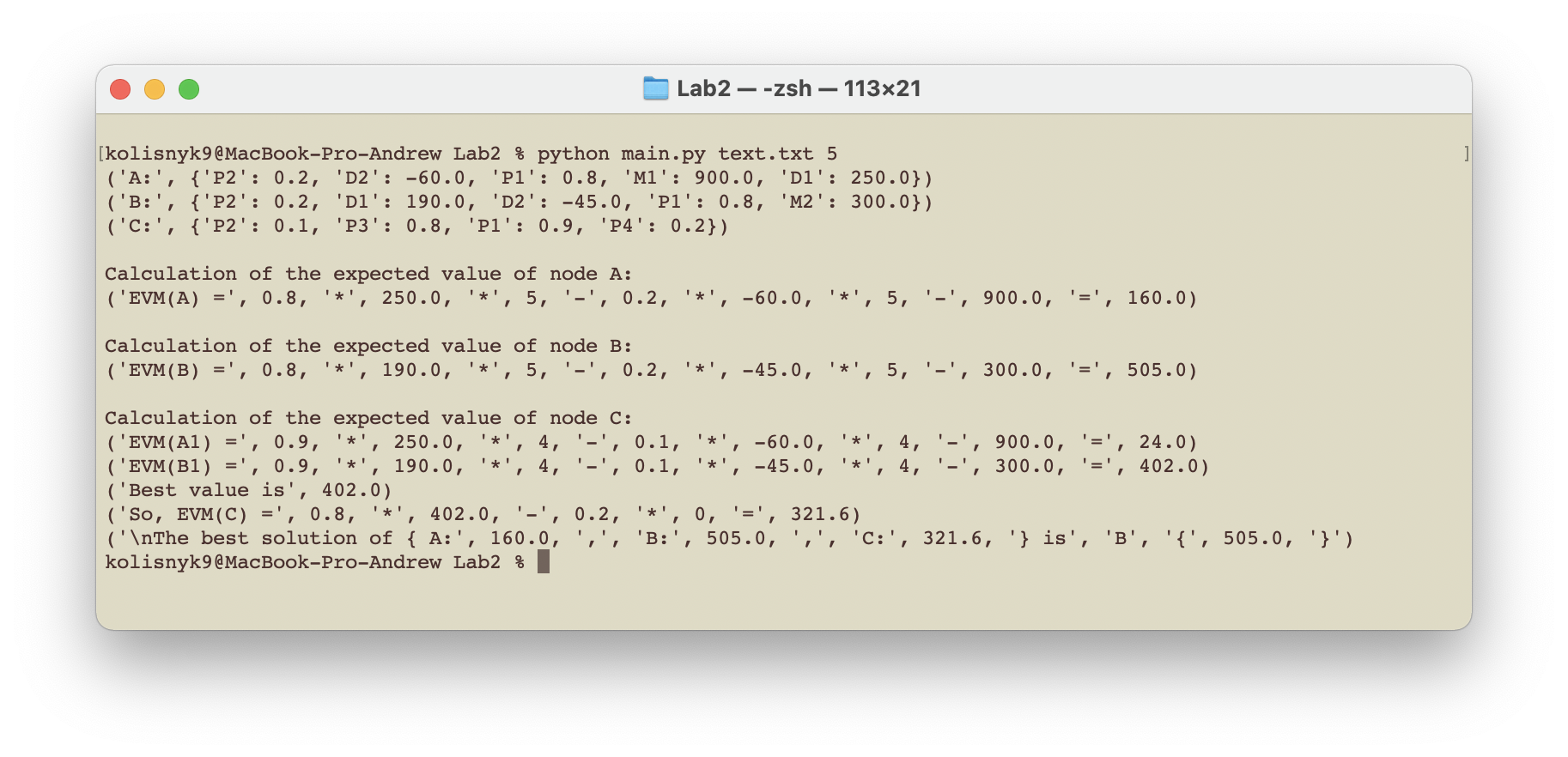


Рис. 4 Відкладення будівництва заводу на 1 рік

З отриманих результатів, можемо зробити висновок, що найкращим буде рішення A (рис. 5)

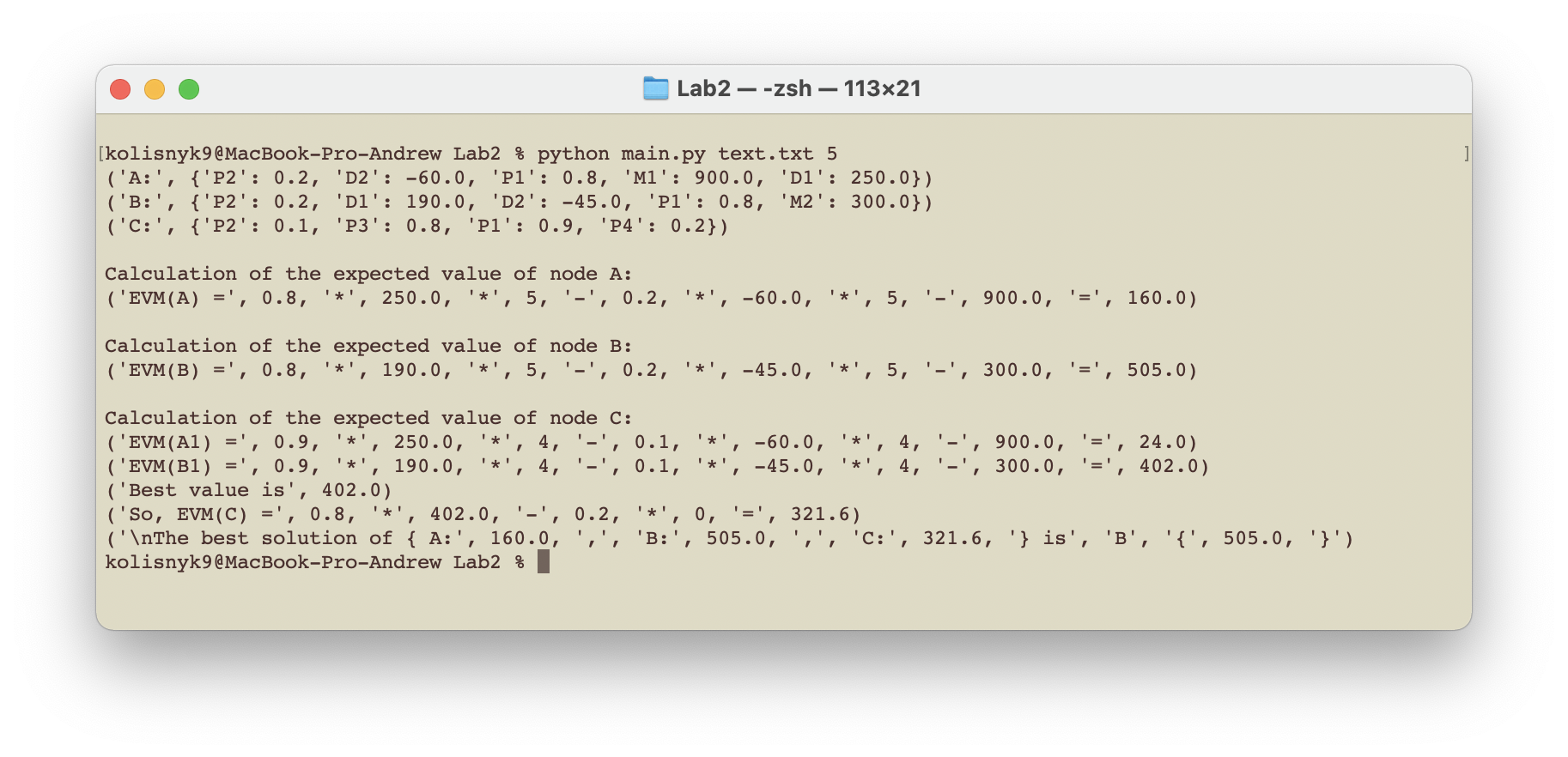


Рис. 5 Отримані результати

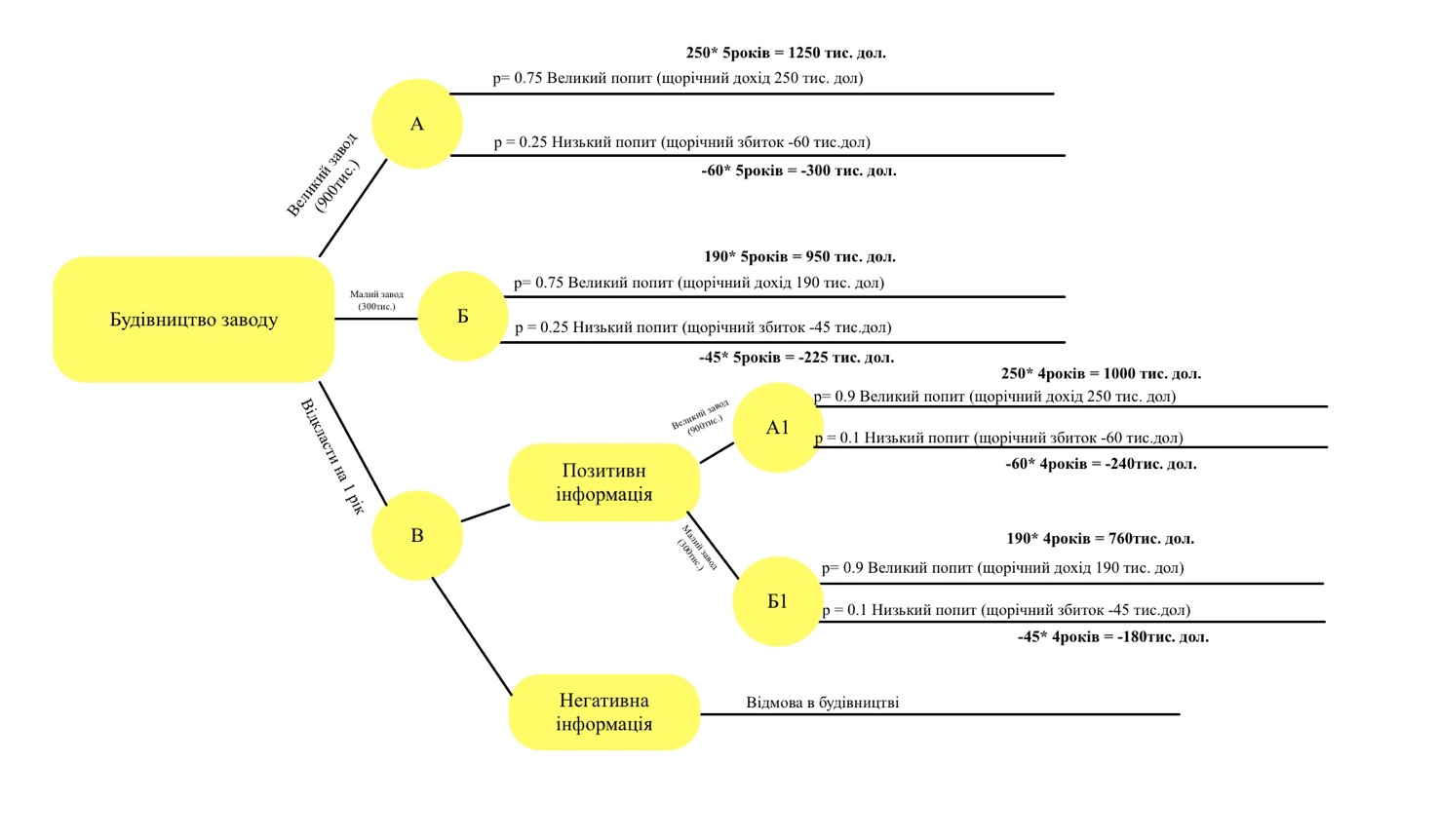


Рис. 6 Дерево рішень

Повний код програми та дерево рішень знаходиться за посиланням:

https://github.com/Kolisnyk9/Lab

**Код програми:**

#!/usr/bin/python

**import** sys

**import** re

**import** string

**def** open\_file():

**if** (len(sys.argv) < 3):

print ('Enter file name and years')

exit()

fileName = sys.argv[1]

years = int(sys.argv[2])

**try**:

**return** [open(fileName), years]

**except** FileNotFoundError:

print("Oops! File not exist...")

exit()

[file, years] = open\_file()

lines = []

benefits = []

**for** line **in** file:

**if** (**not** (line **and** **not** line.isspace()) **or** len(lines) >= 3): **continue**

lines.append(re.split(';', re.sub('\n', '', line)))

**if** len(lines) != 3:

print("Must be 3 lines!")

exit()

**if** len(lines[0]) != 5 **or** len(lines[1]) != 5 **or** len(lines[2]) != 4:

print("Wrong data...")

exit()

A = { 'M1': float(lines[0][0]), 'D1': float(lines[0][1]), 'P1': float(lines[0][2]), 'D2': float(lines[0][3]), 'P2': float(lines[0][4]) }

B = { 'M2': float(lines[1][0]), 'D1': float(lines[1][1]), 'P1': float(lines[1][2]), 'D2': float(lines[1][3]), 'P2': float(lines[1][4]) }

C = { 'P3': float(lines[2][0]), 'P4': float(lines[2][1]), 'P1': float(lines[2][2]), 'P2': float(lines[2][3]) }

print("A:", A)

print("B:", B)

print("C:", C)

print("\nCalculation of the expected value of node A:")

EVM\_A = A['P1'] \* A['D1'] \* years - A['P2'] \* A['D2'] \* years - A['M1'];

print("EVM(A) =", A['P1'], '\*', A['D1'], '\*', years, '-', A['P2'], '\*', A['D2'], '\*', years, '-', A['M1'], '=', EVM\_A)

print("\nCalculation of the expected value of node B:")

EVM\_B = B['P1'] \* B['D1'] \* years - B['P2'] \* B['D2'] \* years - B['M2'];

print("EVM(B) =", B['P1'], '\*', B['D1'], '\*', years, '-', B['P2'], '\*', B['D2'], '\*', years, '-', B['M2'], '=', EVM\_B)

print("\nCalculation of the expected value of node C:")

EVM\_A1 = C['P1'] \* A['D1'] \* (years - 1) - C['P2'] \* A['D2'] \* (years - 1) - A['M1']

EVM\_B1 = C['P1'] \* B['D1'] \* (years - 1) - C['P2'] \* B['D2'] \* (years - 1) - B['M2']

print("EVM(A1) =", C['P1'], '\*', A['D1'], '\*', years - 1, '-', C['P2'], '\*', A['D2'], '\*', years - 1, '-', A['M1'], '=', EVM\_A1)

print("EVM(B1) =", C['P1'], '\*', B['D1'], '\*', years - 1, '-', C['P2'], '\*', B['D2'], '\*', years - 1, '-', B['M2'], '=', EVM\_B1)

EVM\_MAX = max([EVM\_A1, EVM\_B1])

print("Best value is", EVM\_MAX)

EVM\_C = C['P3'] \* EVM\_MAX - C['P4'] \* 0

print("So, EVM(C) =", C['P3'], '\*', EVM\_MAX, '-', C['P4'], '\*', 0, '=', EVM\_C)

letters = ['A', 'B', 'C']

bestSolution = max([EVM\_A, EVM\_B, EVM\_C])

print("\nThe best solution of { A:", EVM\_A, ',', "B:", EVM\_B, ',', "C:", EVM\_C, "} is", letters[[EVM\_A, EVM\_B, EVM\_C].index(bestSolution)], '{', bestSolution, '}')

**Висновок:** в ході виконання лабораторної роботи було отримано теоретичні знання про дерева рішень та їх використання, визначено найкращий варіант будівництва заводу, пораховано потрібні значення вручну, а також написано програму, яка автоматизує даний процес.