

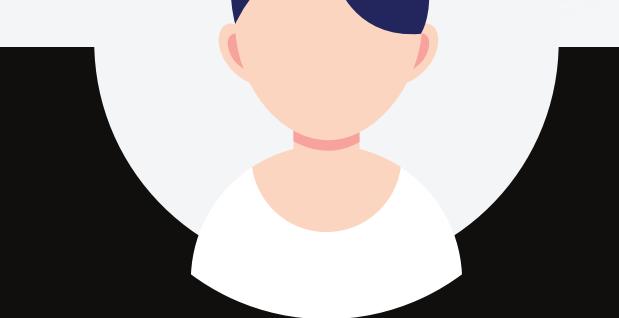


# **PELURUHAN RADIOISOTOP**

DISUSUN OLEH : KELOMPOK 12



# OUR TEAM



Sena  
Patriarka C.  
140310230015



M. Ihsan Putra  
Pratama M.  
140310230024



Falesya  
Luneta A.  
140310230059

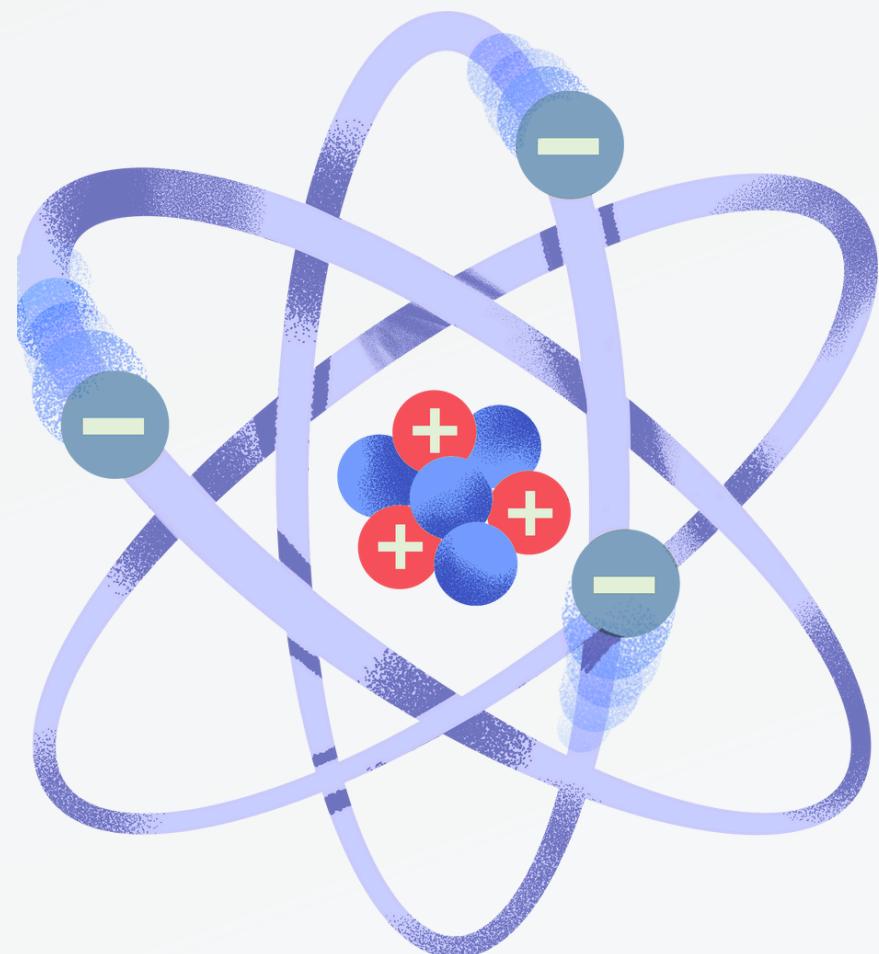


# CONTENT

- 01** LATAR BELAKANG
- 02** TUJUAN
- 03** DESKRIPSI PROGRAM
- 04** MODUL YANG DIGUNAKAN
- 05** FLOWCHART
- 06** LISTINGAN PROGRAM
- 07** KESIMPULAN

# 01

# LATAR BELAKANG

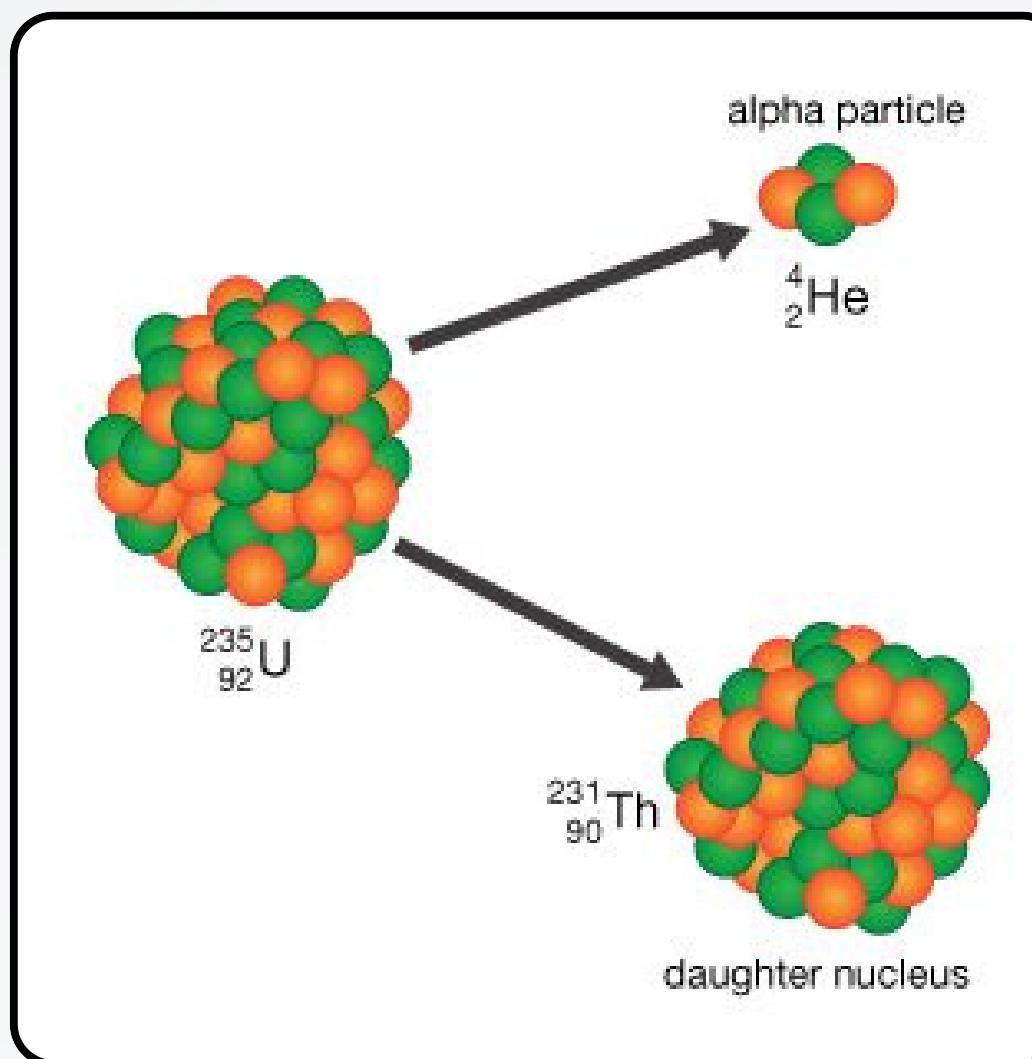


Setiap materi yang ada di alam semesta ini tersusun dari partikel-partikel kecil yang disebut atom, dimana inti atom atau nukleus tersusun dari neutron dan proton dengan elektron tersebar mengorbit terhadap inti tersebut. Jumlah dari proton, neutron, dan elektron akan mempengaruhi fasa dan sifat dari atom tersebut, masing-masing kombinasi dalam atom ini memiliki namanya tersendiri dan disebut sebagai unsur.

Pada umumnya, sebagian besar atom akan berada dalam keadaan stabil, apabila jumlah proton dan neutron dalam nukleus bertambah ataupun berkurang, hal ini akan menyebabkan perubahan energi internal sehingga atom pun menjadi tidak stabil, atom dalam keadaan ini disebut sebagai zat radioaktif .

# 01

# LATAR BELAKANG



Peluruhan zat radioaktif adalah proses alami di mana inti atom yang tidak stabil memancarkan radiasi dan berubah menjadi inti atom lain yang lebih stabil. Proses ini terjadi secara spontan dan tidak dapat diubah. Peluruhan zat radioaktif memiliki banyak aplikasi dalam berbagai bidang, seperti:

- Kedokteran: Digunakan dalam terapi kanker, diagnostik medis, dan pencitraan medis.
- Industri: Digunakan dalam sterilisasi makanan, pelacakan material, dan pengukuran usia benda purba.
- Penelitian ilmiah: Digunakan untuk mempelajari struktur atom, reaksi nuklir, dan asal-usul alam semesta.

# 01

# LATAR BELAKANG

## PELURUHAN

Konstanta Peluruhan ( $\lambda$ )

$$\lambda = \frac{\ln 2}{T_{1/2}} = \frac{0,693}{T_{1/2}}$$

Aktivitas Peluruhan ( $A_t$ )

$$A_t = \lambda \cdot N_t = A_o \cdot e^{-\lambda t} = A_o \left(\frac{1}{2}\right)^{\frac{t}{T_{1/2}}}$$

Keterangan:

$\lambda$  = konstanta laju peluruhan

$T_{1/2}$  = waktu paruh

$t$  = waktu peluruhan

$N_t$  = jumlah zat akhir/sisa

$A_t$  = aktivitas peluruhan pada waktu  $t$

$A_o$  = aktivitas peluruhan pada awal

## Waktu Paruh ( $T_{1/2}$ )

$$N_t = N_o e^{-\lambda t} = N_o \left(\frac{1}{2}\right)^{\frac{t}{T_{1/2}}}$$

Keterangan:

$N_t$  = jumlah zat akhir/sisa

$N_o$  = jumlah zat awal/mula-mula

$t$  = waktu peluruhan

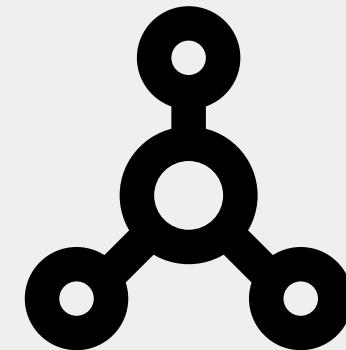
$T_{1/2}$  = waktu paruh

$e$  = bilangan natural = 2,71828

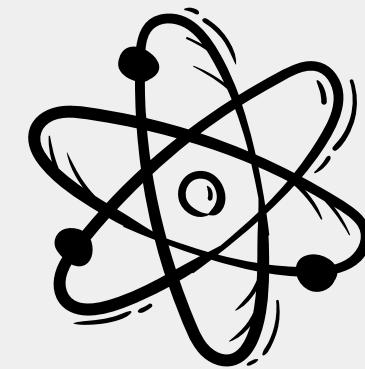
$\lambda$  = konstanta laju peluruhan

02

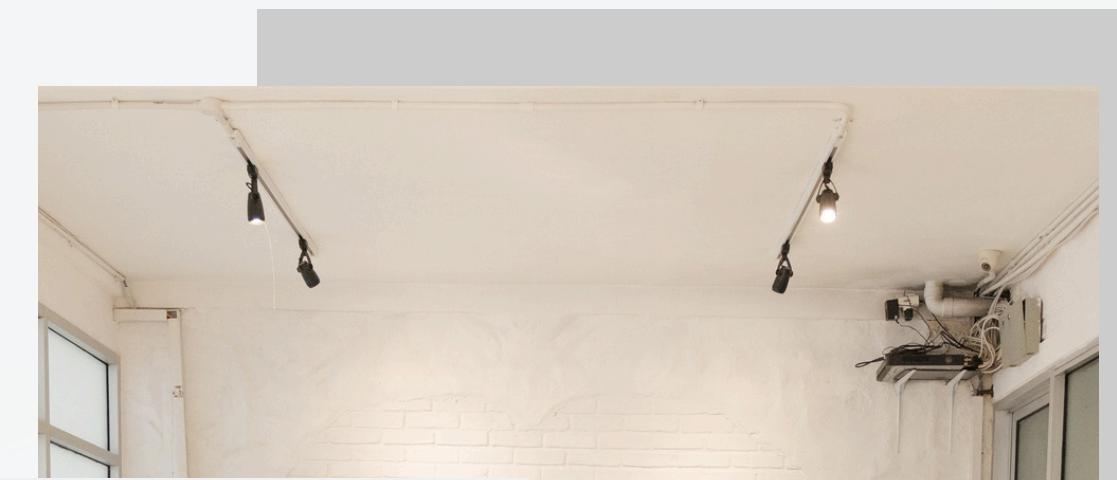
# TUJUAN



Mahasiswa diharapkan dapat mengetahui bagaimana alur peluruhan suatu isotop radioaktif hingga mencapai kestabilan.



Mahasiswa diharapkan dapat mencari jumlah sisa partikel dari waktu yang sudah ditentukan dalam program Peluruhan Unsur Radioaktif.



# 03

# DESKRIPSI PROGRAM

Sesaat setelah dijalankan, program akan menampilkan tiga pilihan menu yang dapat pengguna pilih, yaitu menu peluruhan radioisotop, jalur kestabilan isotop, dan pilihan keluar. Pada menu peluruhan radioisotop, pengguna akan menginput nomor atom yang akan diamati peluruhannya, lamanya waktu peluruhan, dan jumlah awalnya yang dapat berupa jumlah partikel ataupun massa. Selanjutnya, program akan menentukan isotop apa saja yang sesuai dengan nomor atom tersebut. Dari list yang berisi unsur radioaktif beserta paruh waktunya, program akan mengambil paruh waktu yang sesuai dengan nomor atom, lalu program akan menghitung jumlah atom yang tersisa dan memvisualisasikan proses peluruhan sebagai grafik. Pengguna juga dapat menyimpan data yang diperoleh dalam bentuk txt.



# 03

# DESKRIPSI PROGRAM

Kemudian, program akan kembali ke menu awal dan pengguna dapat memilih menu tersebut. Pada menu Jalur kestabilan radioisotop, pengguna akan menginput nomor atom yang akan diamati jalur/deret kestabilannya, lalu program akan menampilkan list dari radioisotop yang sesuai dengan nomor atom tersebut. Selanjutnya, program akan menampilkan grafik yang menunjukkan grafik bagaimana suatu radioisotop dapat mencapai bentuk atom lain yang stabil dan pengguna juga dapat menyimpan data yang diperoleh dalam bentuk txt. Setelah kembali lagi ke menu awal dan pengguna memilih untuk keluar dari program, maka program akan menampilkan suatu pesan dan berhenti beroperasi.



04

# MODUL YANG DIGUNAKAN



## INTERPOLASI

Digunakan untuk menghubungkan titik-titik yang telah diplot terutama untuk menghubungkan isotop-isotop pada menu jalur kestabilan isotop



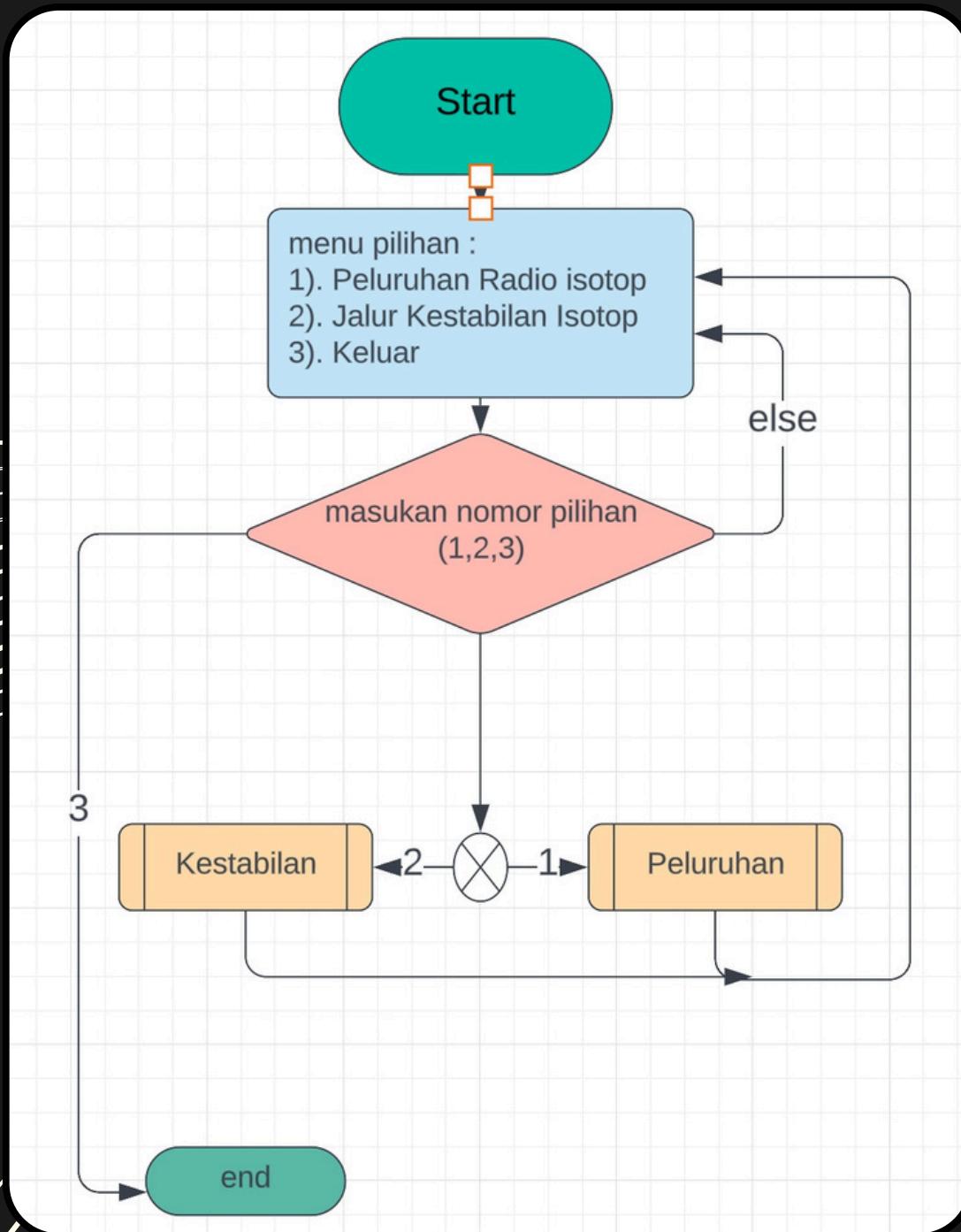
## PDB NUMERIK

Untuk menghitung jumlah zat radioaktif yang masih tersisa pada saat waktu tertentu yang berkelanjutan hingga mencapai waktu luruh

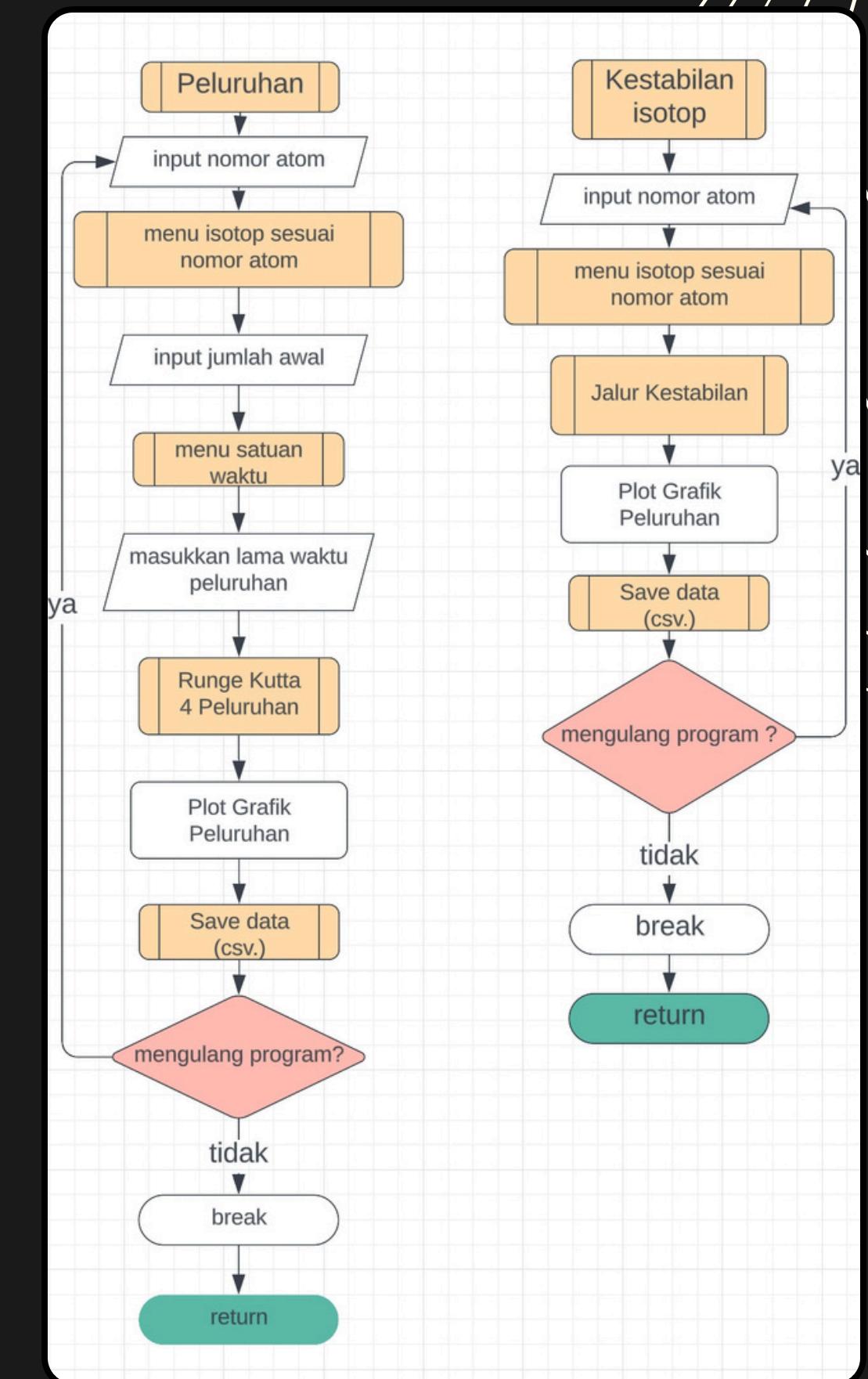
# 05

# FLOWCHART

## PROGRAM UTAMA



## PELURUHAN DAN KESTABILAN



# LISTINGAN PROGRAM

06

```
# MENU DAN PILIHAN
# Menu Pilihan
def MenuPilihan (isi_menu:list, parameter:str, judul_menu:str, keterangan:any) :
    # inisiasi
    n = len(isi_menu)
    banyak_baris = 6
    lebar_nomor = len(f'{n}') + 1

    lebar_teks = []
    for i in range (n) :
        lebar_teks.append(str(isi_menu[i][parameter]))
    lebar_teks = len(max(lebar_teks, key = len)) + 1

    if keterangan == None :
        lebar_keterangan = 0
        keterangan = ''
    else :
        lebar_keterangan = len(str(keterangan))

    lebar_pilihan = 3 + lebar_nomor + lebar_teks + lebar_keterangan
    lebar_menu = 6 + (int((n+1)/banyak_baris) + 1)*(lebar_pilihan)
    maks_lebar_menu = 126
```

# LISTINGAN PROGRAM

06

```
# Pembatasan
if n <= 8 :
    banyak_baris = n
while True :
    if lebar_menu > maks_lebar_menu :
        banyak_baris += 1
        lebar_menu = 6 + (int(n/banyak_baris) + 1)*(lebar_pilihan)
        continue
    elif lebar_menu <= maks_lebar_menu :
        break
```

# LISTINGAN PROGRAM

06

```
# Menampilkan Menu
print(f"[ {judul_menu.upper()} ]".center(lebar_menu, "="))
for kolom in range (banyak_baris) :
    tabel = "|| "
    for baris in range (0, n, banyak_baris) :
        indeks = kolom + baris
        if n == banyak_baris :
            indeks = kolom
        if indeks < n :
            tabel += f'{indeks+1:{lebar_nomor}}. {keterangan:{lebar_keterangan}}{str(isi_menu[indeks][parameter]).capitalize() + ".":{lebar_teks}}'
        elif indeks >= n :
            tabel += f'{"*lebar_pilihan}"'
    print(tabel + " || ")
print(f'{"*lebar_menu}\n')
return
```

# LISTINGAN PROGRAM

06

```
# Pilihan Berbasis Angka
def PilihAngka (pilihan_awal:int, pilihan_akhir:int):
    while True :
        try :
            pilihan = int(input("> Masukkan nomor menu yang diinginkan : "))
            if pilihan not in [i for i in range(pilihan_awal,pilihan_akhir+1)] :
                print("ERROR...Tolong masukkan nomor menu yang sesuai!\n")
                continue
            elif pilihan in [i for i in range(pilihan_awal,pilihan_akhir+1)] :
                return pilihan
        except ValueError :
            print("ERROR...Tolong masukkan input berupa integer!\n")
            continue
```

# LISTINGAN PROGRAM

06

```
# Menyimpan Data sebagai CSV
def SaveData (header : list, data : list, nama_file : str) :
    BanyakListData = len(data)
    BanyakData = len(data[BanyakListData-1])

    with open(nama_file, "w") as DataCSV:
        input_data = csv.writer(DataCSV, delimiter='\t')
        input_data.writerow(header)
        for i in range (BanyakData) :
            baris = []
            for j in range (BanyakListData) :
                baris.append(data[j][i])
            input_data.writerow(baris)

    return
```

# LISTINGAN PROGRAM

06

```
# PROGRAM UTAMA
def main () :
    while True :
        menu = [
            {"Pilihan" : "Peluruhan Radioisotop"},
            {"Pilihan" : "Jalur Kestabilan Radioisotop"},
            {"Pilihan" : "Keluar"}
        ]
        MenuPilihan(menu, "Pilihan", "menu", None)

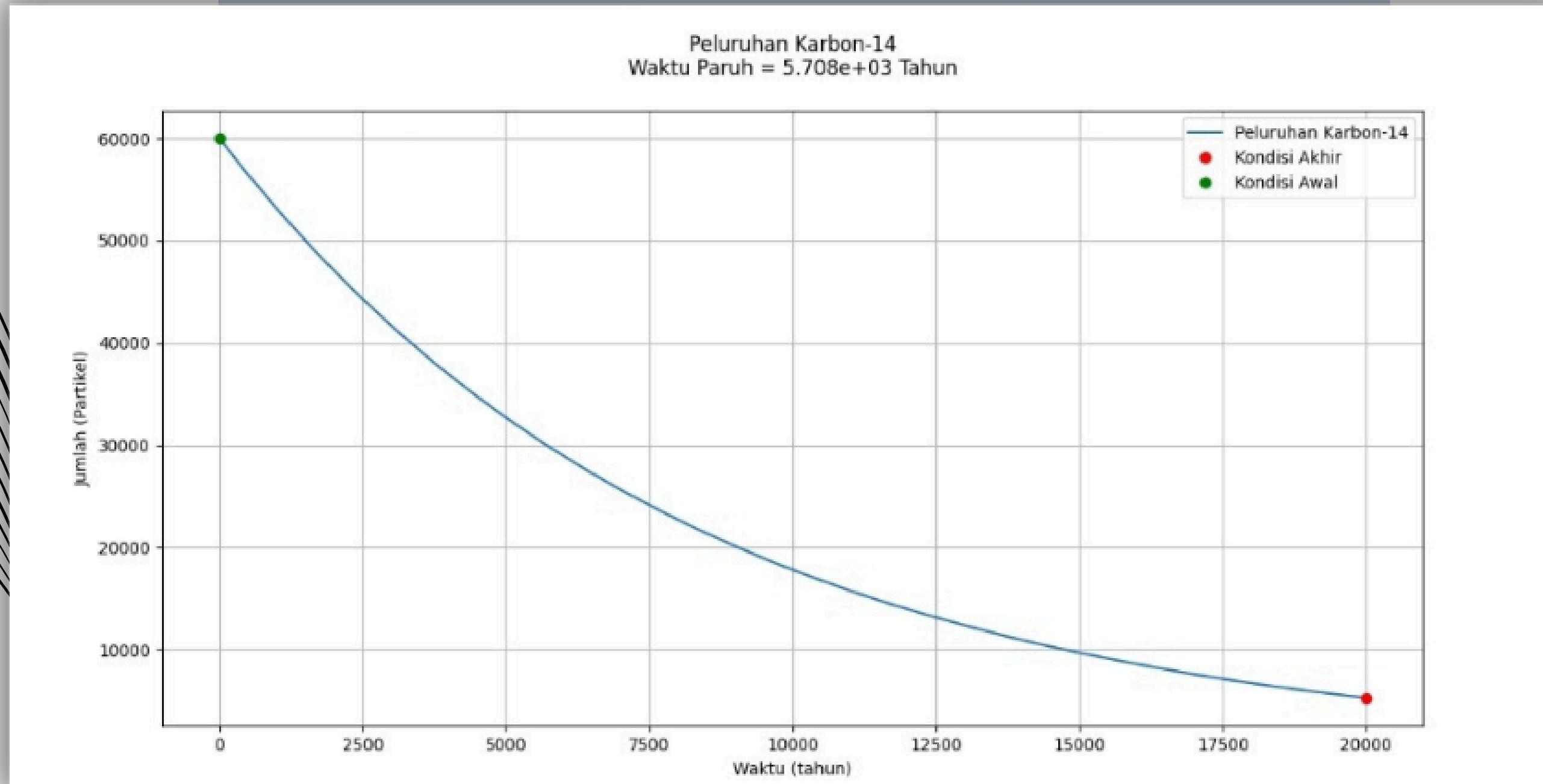
        pilihan = PilihAngka(1,3)
        if pilihan == 1 :
            peluruhan()
            continue
        elif pilihan == 2 :
            kestabilan()
            continue
        elif pilihan == 3 :
            print("\nBaik, terima kasih telah menggunakan program kami...\n")
            break
    return

if __name__ == "__main__":
    main()
```

# HASIL PROGRAM

07

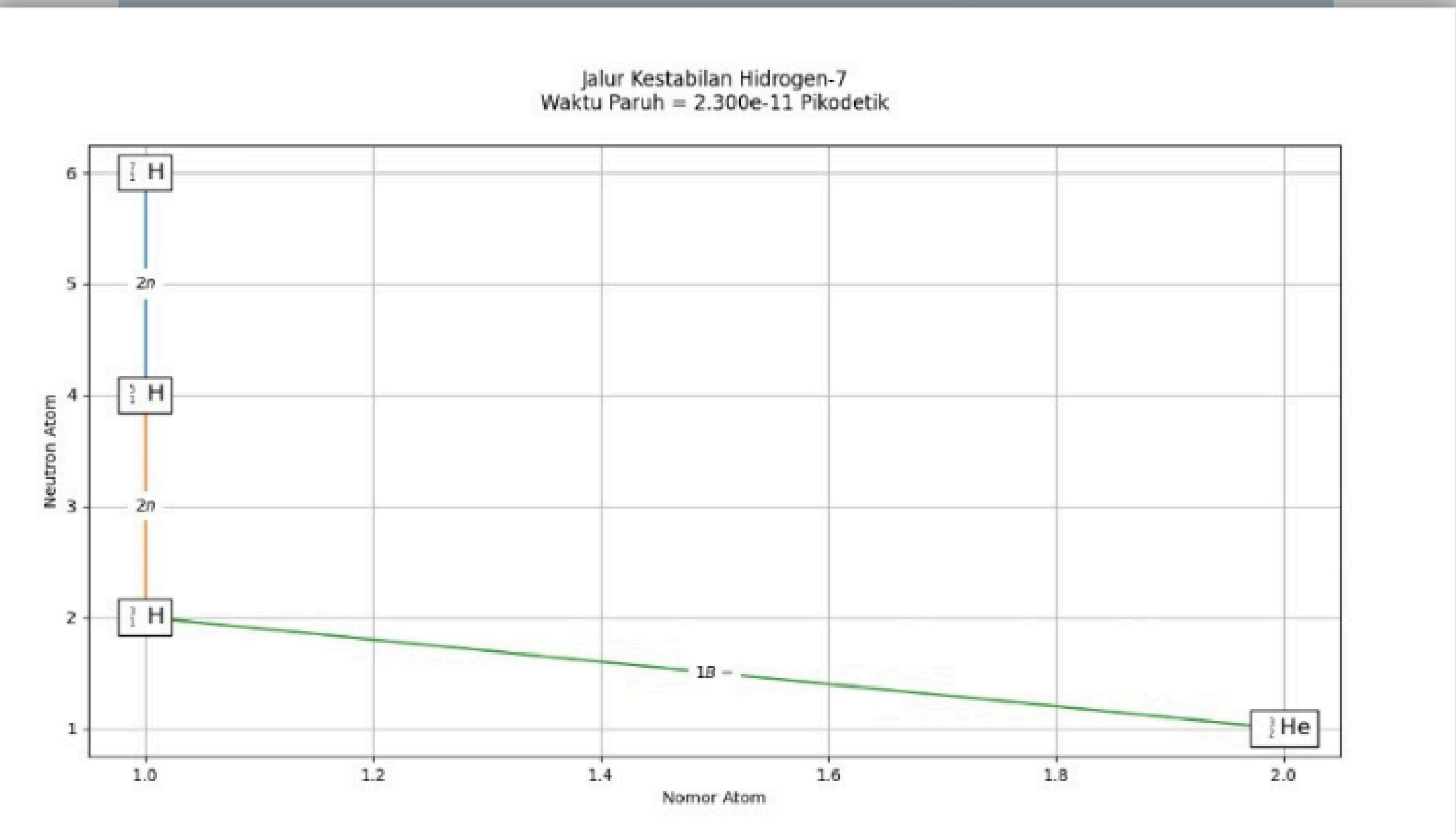
## PADA KARBON-14



# **HASIL PROGRAM**

07

# PADA HIDROGEN-7



**TERIMA  
KASIH**

