**UAS DASPRO TAHUN 2023**

1. FILTER BILANGAN GENAP

|  |
| --- |
| **FILTER BILANGAN GENAP FilterGenap(L)** |
| **DEFINISI DAN SPESIFIKASI**  **FilterGenap** : List -> List  {FilterGenap(L) adalah sebuah fungsi yang menerima list berupa integer dengan output berupa integer genap sesuai inputannya} |
| **REALISASI**  **FilterGenap** (L):  if IsEmpty(L) then  []  else      if FirstElmt(L) % 2 == 0 then         Konso(FirstElmt(L),FilterGenap(Tail(L)))      else         FilterGenap(Tail(L)) |
| **APLIKASI**   * FilterGenap([6,3,1,28,12,9,4]) → [6,28,12,4] |

1. APAKAH BERISI LIST

|  |
| --- |
| **APAKAH BERISI LIST IsContainList (S)** |
| **DEFINISI DAN SPESIFIKASI**  **IsContainList** : List of list → boolean  {IsContainList(S) mengembalikan trus jika salah satu anggota dari list of list S berupa list} |
| **REALISASI**  **IsContainList(S):**  if IsEmpty(S) then     false  else     if IsAtom(FirstList(S)) then        IsContainList(TailList(S))     else        true |
| **REALISASI DALAM PYTHON**  def IsContainList(S):      if IsEmpty(S):          return False      else:          if IsAtom(FirstList(S)):              return IsContainList(TailList(S))          else:              return True |
| **APLIKASI**   * IsContainList([6,[3,1],[28,12,9],4]) → True * IsContainList([6,3,1,28,12,9,4]) → False |

1. MALAS
2. CARI SUB POHON

|  |
| --- |
| **SUB POHON SubTreeElemt (x,P)** |
| **DEFINISI DAN SPESIFIKASI**  **SubTreeElemt** : List of list → List of list  {SubTreeElemt(x,P) memeriksa apakah terdapat sebuah list baru yang memuat seluruh elemen dari subpohon yang dicari, jika huruf atau angka yang dicari tidak ditemukan pada pohon tersebut maka dekembalikan list kosong.} |
| **REALISASI**  **SubTreeElemt (x,P):**  depend on x,P  IsTreeEmpty(P) : []  Akar(P) = x : P  Anak(P) = [] : []  SubTreeELemt(x, Akar(Anak(PN))) : SubTreeElemt(x, Akar(Anak(PN)))  else : SubTreeElemt(x, Konso(Akar(PN), [Tail(Anak(PN))])) |
| **APLIKASI**   * SubTreeElemt(‘K’, ['A',[['K',[['M',[]],['L',[]],['F',[]]]], ['Z',[['I',[]],['R',[]],['Y',[['D',[]]]]]]]]) → ['K',[['M', []],['L',[]],['F',[]]]] |

1. NILAI TERBESAR BAGI 4

|  |
| --- |
| **APAKAH BERISI LIST IsContainList (S)** |
| **DEFINISI DAN SPESIFIKASI**  **IsMaxBST4** : List of list → boolean  {IsMaxBST4(S) memeriksa nilai terbesar pada BST tersebut kemudian mengolah nilai terbesar tersebut apakah dapat dibagi 4 atau tidak dan mengembalikan True pada fungsi tersebut jikanilai terbesar dapat dibagi 4 dan False jika tidak dapat dibagi 4 atau BST yang diberikan kosong}  **F :** integer → boolean  {F (x) adalah fungsi untuk mengecek bilangan tersebut dapat dibagi 4 atau tidak} |
| **REALISASI**  **F :** λ x. x mod 4 = 0  **IsMaxBST4(P,F):**  depend onP,F  IsEmpty(P) : False      IsEmpty(Right(P)) : F(Akar(P))      IsUnerLeft(P) : F(Akar(P))      else : IsMaxBST4(Right(P),F) |
| **APLIKASI**   * IsMaxBST4([50,[17,[12,[],[]],[23,[],[]]],[72,[54,[],[]],[76, [],[]]]],F) → True |