

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN DASAR
POSTTEST 7



Informatika A2'24
Ridho Setiawan
2409106029

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2024

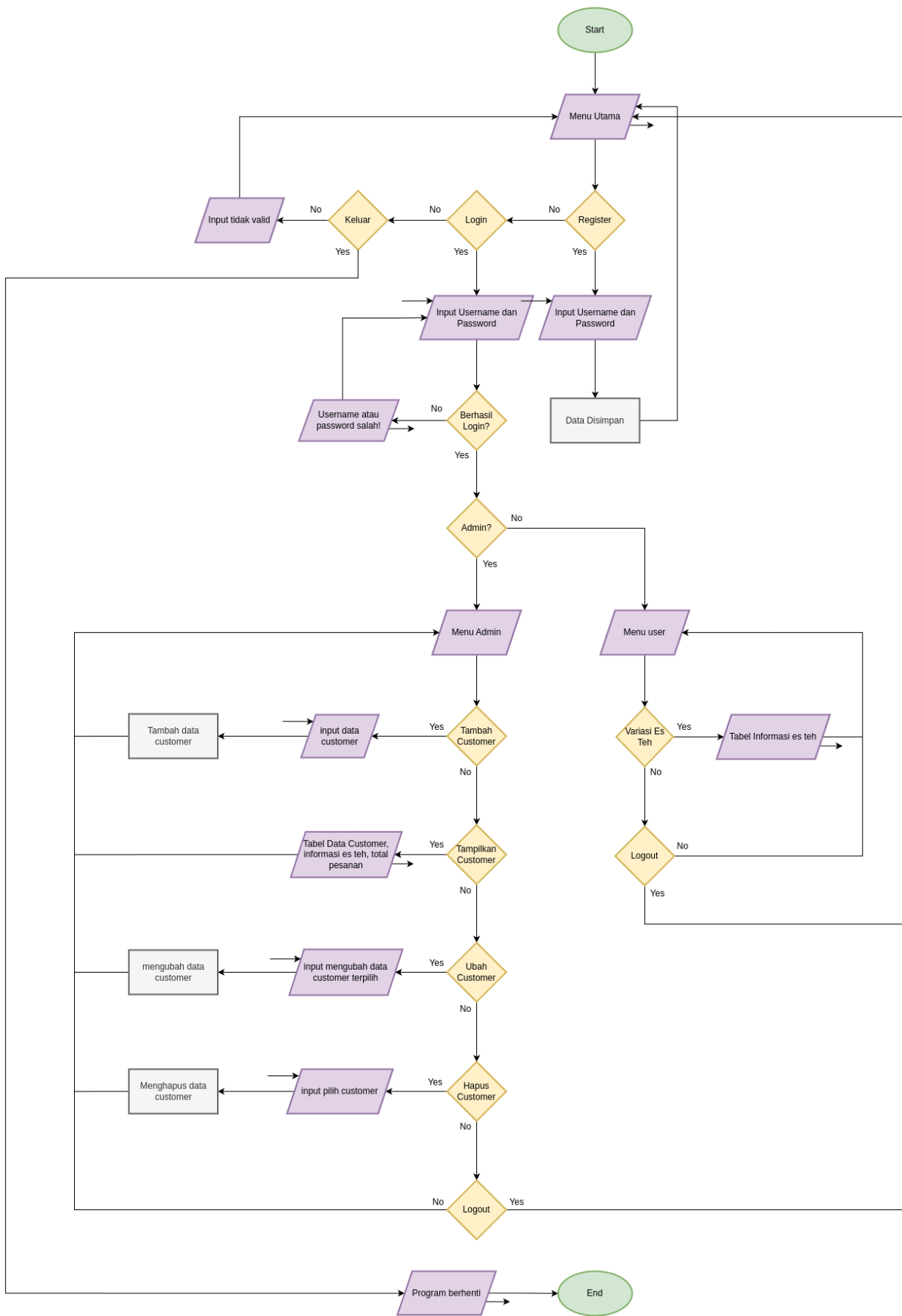
PEMBAHASAN

1.1 LATAR BELAKANG

Program dimulai dengan menu utama memilih ingin langsung login atau buat akun terlebih dahulu, jika login dengan role admin maka admin dapat menambah data customer berupa nama dan ukuran teh, maka program otomatis akan menghitung laba bersih dari tiap penjualan dengan mengurangi harga jual dengan modal, admin juga dapat melihat data customer secara rinci, mengupdate data customer jika terjadi kesalahan input, dan menghapus data customer, jika login dengan role user maka, hanya dapat melihat informasi variasi es teh dan jumlah pemesanan es teh mamat.

Cara menyelesaikan soal posttest adalah dengan membaca ulang terlebih dahulu modul yang sudah dipelajari pada pertemuan sebelumnya, lalu agar lebih terurut dan mempermudah dalam pembuatan program saya awali tugas posttest ini dengan membuat **flowchart** menggunakan website **draw.io**, dan langkah terakhir adalah **pembuatan program** berdasarkan flowchart yang telah dibuat menggunakan bahasa pemrograman python, dan selesai, terimakasih.

1.2 FLOWCHART



1.3 SCREENSHOT CODING

```
1  import os
2  from tabulate import tabulate
3  import time
4  from tqdm import tqdm
5  import random
6
7  # Warna
8  BOLD = '\033[1m'
9  BLACK = '\033[30m'
10 RED = '\033[31m'
11 GREEN = '\033[32m'
12 YELLOW = '\033[33m'
13 BLUE = '\033[34m'
14 MAGENTA = '\033[35m'
15 CYAN = '\033[36m'
16 LIGHT_GRAY = '\033[37m'
17 DARK_GRAY = '\033[90m'
18 BRIGHT_RED = '\033[91m'
19 BRIGHT_GREEN = '\033[92m'
20 BRIGHT_YELLOW = '\033[93m'
21 BRIGHT_BLUE = '\033[94m'
22 BRIGHT_MAGENTA = '\033[95m'
23 BRIGHT_CYAN = '\033[96m'
24 WHITE = '\033[97m'
25 RESET = '\033[0m'
26
27
28 dataAkun = {
29     1: {'username': "admin", "password": "1234#"},
30     0: {
31         1: {"username": "ridho", "password": "029"},
32     }
33 }
34
35 jenisEsteh = {
36     "S": {"ukuran": "Small", "modal": 2000, "harga": 3000},
37     "M": {"ukuran": "Medium", "modal": 3500, "harga": 5000},
38     "J": {"ukuran": "Jumbo", "modal": 6000, "harga": 8000}
39 }
```

```
40
41 data_customers = {}
42
43 # clear terminal
44 def clear():
45     os.system("cls || clear")
46
47 # fake loading
48 def loading(loadingLength, delay):
49     for i in tqdm(range(loadingLength)):
50         time.sleep(delay)
51
52 # fake loading (2)
53 random = random.randint(1, 2)
54 def wait(index):
55     clear()
56     print("Loading", end='')
57     for i in range(3):
58         print(".", end='', flush=True)
59         time.sleep(0.2)
60
61     if index < random:
62         wait(index+1)
63
64 # validasi input bertipe string
65 def validasi_input_huruf(charInput):
66     for char in charInput:
67         if not (char.isalpha()):
68             return False
69     return True
70
71 # validasi input bertipe number
72 def validasi_input_angka(numInput):
73     return numInput.isdigit()
74
75 # menahan dan melanjutkan program
76 def lanjut():
77     print("Tekan enter untuk lanjut...")
78     input()
```

```

79     clear()
80
81     # mengubah nama ukuran teh
82     def nama_ukuran_teh(ukuran):
83         if ukuran in ("S", "M", "J"):
84             if ukuran == "S":
85                 ukuran = "Small"
86             elif ukuran == "M":
87                 ukuran = "Medium"
88             elif ukuran == "J":
89                 ukuran = "Jumbo"
90         return ukuran
91
92     # tambah data customer
93     def tambahCustomer():
94         clear()
95         print("====+ TAMBAH DATA +====\n")
96
97         index_data_customers = (len(data_customers))
98
99         while True:
100             try:
101                 nama = input("Nama customer: ").strip()
102                 if not nama:
103                     raise ValueError("Nama customer tidak boleh kosong!")
104                 if not validasi_input_huruf(nama):
105                     raise ValueError("Nama customer hanya bisa berisi huruf!")
106
107                 ukuranTeh = input("Ukuran Teh (S/M/J): ").upper().strip()
108                 if not ukuranTeh:
109                     raise ValueError("Ukuran teh tidak boleh kosong!")
110                 if not validasi_input_huruf(ukuranTeh):
111                     raise ValueError("Ukuran teh hanya bisa berisi huruf!")
112                 if ukuranTeh not in ("S", "M", "J"):
113                     raise ValueError("Hanya bisa (S/M/J)")
114
115                 ukuranTeh = nama_ukuran_teh(ukuranTeh)
116
117                 for teh in jenisEsteh.values():
118                     if ukuranTeh == teh["ukuran"]:
119                         hargaJual = teh["harga"]
120                         labaBersih = hargaJual - teh["modal"]
121
122                 index_data_customers += 1
123
124                 data_customers[index_data_customers] = {"nama": nama, "ukuran_teh": ukuranTeh, "harga_jual": hargaJual, "laba_bersih": labaBersih}
125
126                 delay = 0.1
127                 if index_data_customers > 15:
128                     delay = 0.07
129                 elif index_data_customers > 30:
130                     delay = 0.05
131                 elif index_data_customers > 45:
132                     delay = 0.03
133                 elif index_data_customers > 60:
134                     delay = 0.01
135
136                 loading(index_data_customers, delay)
137
138                 print(GREEN+BOLD+"\nBerhasil menambah data customer!" +RESET)
139                 lanjut()
140                 break
141
142             except ValueError as error:
143                 print(f"{RED}{BOLD}Input tidak valid: {error} Silakan coba lagi.\n{RESET}")
144
145     # tabel rincian pesanan
146     def rincianPesanan():
147         totalS = 0
148         totalM = 0
149         totalJ = 0
150
151         for data in data_customers.values():
152             if data["ukuran_teh"] == 'Small':
153                 totalS += 1

```

```

154         elif data["ukuran_teh"] == 'Medium':
155             totalM += 1
156         elif data["ukuran_teh"] == 'Jumbo':
157             totalJ += 1
158
159     totalPemesanan = 0
160     totalPemesanan += len(data_customers)
161
162     table = [{"Small", "280ml", totalS}, {"Medium", "400ml", totalM}, {"Jumbo", "600ml", totalJ}, {"Total", "", totalPemesanan}]
163     headers = ["Ukuran", "Isi bersih", "total pesanan"]
164
165     print(BOLD+"\nTabel Rincian Pesanan"+RESET+BLUE)
166     print(tabulate(table, headers, tablefmt="grid")+RESET)
167
168     # tampilkan data customer
169     def tampilkanCustomer():
170         clear()
171         print("==== DATA CUSTOMER ====")
172
173         if not data_customers:
174             print(RED+BOLD+"\nBelum ada Customer"+RESET)
175             time.sleep(1)
176         else:
177             totalKeuntungan = 0
178
179             table = []
180
181             for i, data in enumerate(data_customers.values(), start=1):
182                 row = [i] + list(data.values())
183                 table.append(row)
184                 headers = ["No"] + list(data.keys())
185
186                 totalKeuntungan += data["laba_bersih"]
187
188             # Menampilkan tabel data customer
189             print(BOLD+"\nTabel Data Customer"+RESET+BLUE)

```

```

190     print(tabulate(table, headers, tablefmt="grid")+RESET)
191
192     # Menampilkan rincian pesanan
193     rincianPesanan()
194
195     # Menampilkan total keuntungan
196     print(f"{BOLD}\nTotal Keuntungan : Rp{totalKeuntungan}\n(RESET}")
197     lanjut()
198
199
200     # update data customer
201     def updateCustomer():
202         clear()
203         print("==== UPDATE ====")
204         if not data_customers:
205             print(RED+BOLD+"\nBelum ada Customer"+RESET)
206             time.sleep(1)
207         else:
208             print()
209             for i, data in enumerate(data_customers.values()):
210                 print(f"{BLUE}{BOLD}{i+1}. {data['nama']}(RESET}")
211
212             while True:
213                 try:
214                     pilihCustomer = (input("\nPilih customer yang akan di-update datanya (1/2/3...): ")).strip()
215
216                     if not pilihCustomer:
217                         raise ValueError("Tidak boleh kosong!")
218                     if not validasi_input_angka(pilihCustomer):
219                         raise ValueError("Hanya bisa angka!")
220
221                     pilihCustomer = int(pilihCustomer)
222
223                     if pilihCustomer not in range(1, len(data_customers) + 1):
224                         raise ValueError("Data customer tidak ditemukan")
225

```

```

226 # Mengambil data berdasarkan indeks
227 dataTerpilih = list(data_customers.values())[pilihCustomer - 1]
228 break
229
230 except ValueError as error:
231     print(f'{RED}{BOLD}Input tidak valid: {error} Silakan coba lagi.{RESET}')
232
233     print(f'''{BLUE}{BOLD}
234 | Nama: {dataTerpilih["nama"]}
235 | Ukuran Teh: {dataTerpilih["ukuran_teh"]}
236 | Harga Jual: Rp{dataTerpilih["harga_jual"]}
237 | Keuntungan: Rp{dataTerpilih["laba_bersih"]}
238 {RESET}''')
239
240 while True:
241     try:
242         updateNama = input("\nUbah nama (tekan enter jika tidak ingin mengubah: ").strip()
243         if not validasi_input_huruf(updateNama):
244             raise ValueError("Nama customer hanya bisa berisi huruf!")
245
246         updateUkTeh = input("\nUbah ukuran teh (S/M/J) (tekan enter jika tidak ingin mengubah: ").upper().strip()
247         if not validasi_input_huruf(updateUkTeh):
248             raise ValueError("Ukuran teh hanya bisa berisi huruf!")
249         if updateUkTeh and updateUkTeh not in ("S", "M", "J"):
250             raise ValueError("Hanya bisa (S/M/J)")
251
252         updateUkTeh = nama_ukuran_teh(updateUkTeh)
253
254         # Update data jika ada input baru
255         update_by_index = list(data_customers.values())[pilihCustomer-1]
256         print(update_by_index)
257         if updateNama:
258             update_by_index["nama"] = updateNama
259         if updateUkTeh:
260             update_by_index["ukuran_teh"] = updateUkTeh
261
262         # Hitung ulang harga jual dan laba bersih berdasarkan ukuran teh baru
263         ukuranTeh = update_by_index["ukuran_teh"] # Ambil ukuran teh terbaru

```

```

264     for teh in jenisEsteh.values():
265         if ukuranTeh == teh["ukuran"]:
266             update_by_index["harga_jual"] = teh["harga"]
267             update_by_index["laba_bersih"] = teh["harga"] - teh["modal"]
268
269     loading(10, 0.1)
270     print()
271     print(GREEN+BOLD+"\nBerhasil update data customer!" +RESET)
272     lanjut()
273     break
274
275 except ValueError as error:
276     print(f'{RED}{BOLD}Input tidak valid: {error} Silakan coba lagi.\n{RESET}')
277
278 # hapu data customer
279 def hapusCustomer():
280     clear()
281     print("==== DELETE ====")
282     if not data_customers:
283         print(RED+BOLD+"\nBelum ada Customer" +RESET)
284         time.sleep(1)
285     else:
286         print()
287         for i, data in enumerate(data_customers.values()):
288             print(f'{RED}{BOLD}{i+1}. {data["nama"]} {RESET}')
289
290     while True:
291         try:
292             inputHapus = (input("\nData customer yang ingin dihapus (1/2/3...) : ").strip()
293             if not inputHapus:
294                 raise ValueError("Tidak boleh kosong!")
295             if not validasi_input_angka(inputHapus):
296                 raise ValueError("Hanya bisa angka!")
297
298             inputHapus = int(inputHapus)
299
300             hapus_by_index = list(data_customers)[inputHapus-1]
301             del data_customers[hapus_by_index]

```



```

303         loading(10, 0.1)
304         print()
305         print(GREEN+BOLD+"Customer berhasil dihapus"+RESET)
306         lanjut()
307         break
308
309     except ValueError as error:
310         print(f"{RED}{BOLD}Input tidak valid: {error} Silakan coba lagi.\n{RESET}")
311
312 # menu untuk admin
313 def menuAdmin():
314     wait(0)
315     print(RED+BOLD+"\nLogin sebagai admin 🍷"+RESET)
316     time.sleep(1.5)
317     clear()
318     while True:
319         print(BOLD+BLUE+'''
320 KELOLA COSTUMER
321
322 > 1. Tambah Customer
323 > 2. Tampilkan Customer
324 > 3. Ubah Customer
325 > 4. Hapus Customer
326 > 0. Logout
327
328 '''+RESET)
329         pilih = input("Pilih (0-4): ")
330
331         if pilih == "1":
332             clear()
333             tambahCustomer()
334         elif pilih == "2":
335             clear()
336             tampilkanCustomer()
337         elif pilih == "3":
338             clear()

```

```

339         updateCustomer()
340         elif pilih == "4":
341             clear()
342             hapusCustomer()
343         elif pilih == "0":
344             clear()
345             print("Logout")
346             time.sleep(1)
347             menuUtama()
348         else:
349             clear()
350             print(RED+BOLD+"input tidak valid!\n"+RESET)
351
352 # menu untuk user
353 def menuUser():
354     wait(0)
355     print(RED+BOLD+"\nLogin sebagai user 🍷"+RESET)
356     time.sleep(1.5)
357     clear()
358     while True:
359         print(BLUE+BOLD+'''
360 PENJUALAN ES TEH MAMAT
361
362 > 1. Variasi Es Teh
363 > 0. Logout
364
365 '''+RESET)
366         pilih = input("Pilih (0-1): ")
367
368         if pilih == "1":
369             clear()
370             rincianPesanan()
371             print()
372             lanjut()
373         elif pilih == "0":
374             clear()
375             print("Logout")

```



```

451         if not login_sukses:
452             raise ValueError("Username atau password salah, silakan coba lagi!")
453
454         break # Keluar dari loop utama jika login sukses
455
456     except ValueError as error:
457         print(f"{RED}{BOLD}Input tidak valid: {error} Silakan coba lagi.\n{RESET}")
458
459
460 # menu utama es teh mamat
461 def menuUtama():
462     clear()
463     print(BOLD+BLUE+
464 '''
465
466
467
468
469
470
471
472 > 1. Register
473 > 2. Login
474 > 0. Keluar
475
476 ''' + RESET)
477     while True:
478         inputMenuUtama = input("Pilih (0-2) : ")
479
480         if inputMenuUtama == "1":
481             registrasi()
482             menuUtama()
483         elif inputMenuUtama == "2":
484             login()
485         elif inputMenuUtama == "0":
486             print("Program Berhenti")
487             exit(0)
488         else:
489             print(RED+BOLD+"input tidak valid!\n"+RESET)
490
491     menuUtama()

```