

**LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA**

**UTS**



**NAMA: ALFIN AFRIANSYAH**

**NIM: 2341760089**

**PRODI: D-IV SISTEM INFORMASI BISNIS**

**KELAS: 1E**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

Berikut kode program untuk soal UTS

```
1 public class Transaksi03 {
2     double saldo;
3     double saldoAwal;
4     double saldoAkhir;
5     String tanggalTransaksi;
6     String type;
7     String noRekening;
8
9     public Transaksi03(Rekening03 rekening, double a, double b, double c, String d, String e){
10         saldo = a;
11         saldoAwal = b;
12         saldoAkhir = c;
13         tanggalTransaksi = d;
14         type = e;
15         this.noRekening = rekening.noRekening;
16     }
17
18     void tampilkanDataSaldo() {
19         System.out.printf("%-20s%-15s%-20s%-15s%-25s\n",noRekening, saldo, saldoAwal, saldoAkhir, tanggalTransaksi, type);
20     }
21 }
```

```
1 public class Rekening03 {
2     String noRekening;
3     String nama;
4     String namaIbu;
5     String phone;
6     String email;
7
8     public Rekening03(String a, String b, String c, String d, String e){
9         noRekening = a;
10        nama = b;
11        namaIbu = c;
12        phone = d;
13        email = e;
14    }
15
16    void tampilkanData(){
17        System.out.printf("%-15s%-20s%-20s%-15s%-25s\n", noRekening, nama, namaIbu, phone, email);
18        System.out.println();
19    }
20 }
```

```

1 public class Service03 {
2     Rekening03 list [] = new Rekening03[19];
3     Transaksi03 list2 [] = new Transaksi03[19];
4     int idx1, idx2;
5     void tambah1 (Rekening03 M){
6         if (idx1<list.length){
7             list[idx1] = M;
8             idx1++;
9         } else {
10             System.out.println("Data sudah penuh!!");
11         }
12     }
13     void tambah2 (Transaksi03 T){
14         if (idx2<list2.length){
15             list2[idx2] = T;
16             idx2++;
17         } else {
18             System.out.println("Data sudah penuh!!");
19         }
20     }
21
22     void tampil(){
23         System.out.printf("%-15s%-20s%-20s%-15s%-25s\n", "No Rekening", "Nama", "Nama Ibu", "Phone", "Email");
24         for(Rekening03 M : list){
25             M.tampilkanData();
26         }
27     }
28
29     void tampil2(){
30         System.out.printf("%-20s%-15s%-20s%-20s%-15s\n", "No Rekening", "saldo", "saldoAwal", "saldoAkhir", "tanggalTransaksi");
31         for(Transaksi03 T : list2){
32             T.tampilkanDataSaldo();
33         }
34     }
35
36     void tampilkanSaldoDiatas500000() {
37         System.out.println("Daftar No Rekening dengan Saldo di atas 500000:");
38         System.out.println("-----");
39         for (Transaksi03 T : list2) {
40             if (T.saldoAkhir > 500000) {
41                 System.out.printf("%-15s%-15.2f\n", T.noRekening, T.saldoAkhir);
42             }
43         }
44     }
45
46     void sortingName() {
47         for (int i = 0; i < idx1 - 1; i++) {
48             for (int j = 0; j < idx1 - i - 1; j++) {
49                 if (list[j].nama.compareToIgnoreCase(list[j + 1].nama) > 0 ) {
50                     Rekening03 temp = list[j];
51                     list[j] = list[j + 1];
52                     list[j + 1] = temp;
53                 }
54             }
55         }
56     }
57 }

```

```

1  import java.util.Scanner;
2
3  public class MainTransaksi03 {
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6          Service03 list = new Service03();
7          Rekening03 r1 = new Rekening03("16030927 3084","Wallace","Chase Castro","1-458-264-3263","ligula.Nullam@tacitisociosqu.edu");
8          Rekening03 r2 = new Rekening03("16100617 0573","Darius","Julian Daniel","1-357-843-0547","nec@lectusjusto.org");
9          Rekening03 r3 = new Rekening03("16240401 2243","Fuller","Finn Dunlap","571-70623","convallis@Vestibulumanteipsum.org");
10         Rekening03 r4 = new Rekening03("16270525 0112","Malcolm","Keane Floyd","623-0234","porttitor.tellus.non@Curabitur.ca");
11         Rekening03 r5 = new Rekening03("16971204 2416","Geoffrey","Stephen Pratt","1-683-416-8323","ut.pellentesque@luctusutpellentesque.com");
12         Rekening03 r6 = new Rekening03("16100727 8862","Rudyard","Charles Morales","650-5379","Proin.eget@velitegestaslacinia.ca");
13         Rekening03 r7 = new Rekening03("16460329 4259","Troy","Damon Guerra","897-7608","pede.Suspendisse.dui@a.ca");
14         Rekening03 r8 = new Rekening03("16320421 3437","Alec","Cooper Lee","792-4447","non@mus.com");
15         Rekening03 r9 = new Rekening03("16180729 7229","Walter","Seth Drake","863-8209","Pellentesque.ut.ipsum@neque.ca");
16         Rekening03 r10 = new Rekening03("16950313 6823","Simon","Burton Gates","592-6919","tellus.justo.sit@commodoauctor.net");
17         Rekening03 r11 = new Rekening03("16850708 3528","Kamal","Odysseus Salas","1-115-339-7678","dictum@nec.edu");
18         Rekening03 r12 = new Rekening03("16080205 9329","Xenos","Colin Curry","1-891-703-2664","vel@ullamcorper magna.co.uk");
19         Rekening03 r13 = new Rekening03("16080628 2695","Merritt","Clarke Roman","1-978-632-5110","Aliquam.gravida@vestibulumMauris.net");
20         Rekening03 r14 = new Rekening03("16130909 2979","Ryder","Joel Cunningham","817-1766","interdum.Curabitur.dictum@rutrumurna.edu");
21         Rekening03 r15 = new Rekening03("16890212 8688","Preston","Brock Schroeder","1-675-400-4501","et.ultrices@a.co.uk");
22         Rekening03 r16 = new Rekening03("16141008 9963","Alec","Baker Barton","527-9085","ut@aultriciesadipiscing.ca");
23         Rekening03 r17 = new Rekening03("16511222 7763","Price","Ashton Burke","1-564-419-4285","Proin.velit@Ouisac.net");
24         Rekening03 r18 = new Rekening03("16720623 0943","Devin","Slade Flores","977-6690","ac@nibhAliquamornare.com");
25         Rekening03 r19 = new Rekening03("16771126 7372","Ignatius","Lionel Kane","353-5137","cubilia.Curiae.Phaseillus@Ouis.com");
26         list.tambah1(r1);
27         list.tambah1(r2);
28         list.tambah1(r3);
29         list.tambah1(r4);
30         list.tambah1(r5);
31         list.tambah1(r6);
32         list.tambah1(r7);
33         list.tambah1(r8);
34         list.tambah1(r9);
35         list.tambah1(r10);
36         list.tambah1(r11);
37         list.tambah1(r12);
38         list.tambah1(r13);
39         list.tambah1(r14);
40         list.tambah1(r15);
41         list.tambah1(r16);
42         list.tambah1(r17);
43         list.tambah1(r18);
44         list.tambah1(r19);
45         Service03 list2 = new Service03();
46         Transaksi03 s1 = new Transaksi03( r1,898214.0,494048.0,3587.0,"2021-03-09 07:54:42","");
47         Transaksi03 s2 = new Transaksi03( r2, 205420.0,200162.0,775880.0,"2021-06-25 10:23:00","");
48         Transaksi03 s3 = new Transaksi03( r3, 838632.0,350929.0,328316.0,"2021-09-18 23:00:04","");
49         Transaksi03 s4 = new Transaksi03(r4,230659.0,204434.0,690503.0,"2022-02-02 19:10:34","");
50         Transaksi03 s5 = new Transaksi03(r5,770592.0,334245.0,444267.0,"2020-08-11 13:36:56","");
51         Transaksi03 s6 = new Transaksi03(r6,685302.0,451002.0,376442.0,"2020-05-23 07:34:53","");
52         Transaksi03 s7 = new Transaksi03(r7,816129.0,851403.0,597842.0,"2021-07-18 19:41:20","");
53         Transaksi03 s8 = new Transaksi03(r8,989609.0,333823.0,802752.0,"2022-02-01 01:13:11","");
54         Transaksi03 s9 = new Transaksi03(r9,297103.0,396116.0,779589.0,"2021-07-03 01:09:49","");
55         Transaksi03 s10 = new Transaksi03(r10,66190.0,259150.0,619774.0,"2021-09-09 03:57:30","");
56         Transaksi03 s11 = new Transaksi03(r11,234301.0,278309.0,547922.0,"2021-08-24 13:18:39","");
57         Transaksi03 s12 = new Transaksi03(r12,243306.0,869468.0,50283.0,"2021-03-12 03:40:16","");
58         Transaksi03 s13 = new Transaksi03(r13,371045.0,991242.0,602034.0,"2021-08-06 11:48:59","");
59         Transaksi03 s14 = new Transaksi03(r14,395170.0,97058.0,472273.0,"2021-05-02 10:53:31","");
60         Transaksi03 s15 = new Transaksi03(r15,862731.0,561908.0,109431.0,"2021-07-31 08:11:00","");
61         Transaksi03 s16 = new Transaksi03(r16,556798.0,31387.0,725426.0,"2021-03-27 06:18:20","");
62         Transaksi03 s17 = new Transaksi03(r17,873982.0,896213.0,846142.0,"2021-07-18 04:06:30","");
63         Transaksi03 s18 = new Transaksi03(r18,774247.0,739406.0,775848.0,"2020-10-24 01:39:00","");
64         Transaksi03 s19 = new Transaksi03(r19,66987.0,823014.0,868772.0,"2020-12-21 05:57:59","");

```

```

1      list2.tambah2(s1);
2      list2.tambah2(s2);
3      list2.tambah2(s3);
4      list2.tambah2(s4);
5      list2.tambah2(s5);
6      list2.tambah2(s6);
7      list2.tambah2(s7);
8      list2.tambah2(s8);
9      list2.tambah2(s9);
10     list2.tambah2(s10);
11     list2.tambah2(s11);
12     list2.tambah2(s12);
13     list2.tambah2(s13);
14     list2.tambah2(s14);
15     list2.tambah2(s15);
16     list2.tambah2(s16);
17     list2.tambah2(s17);
18     list2.tambah2(s18);
19     list2.tambah2(s19);
20
21     int pilih;
22     System.out.println("-:UTS:-");
23     System.out.println("1. Tampil data rekening");
24     System.out.println("2. Tampil data transaksi");
25     System.out.println("3. Mencari saldo > 500000");
26     System.out.println("4. Sorting by name");
27     System.out.println("5. Keluar");
28     System.out.print("Pilih (1-5): ");
29     pilih = sc.nextInt();
30
31     switch(pilih){
32         case 1:
33             list.tampil();
34             break;
35
36         case 2:
37             list2.tampil2();
38             break;
39
40         case 3:
41             list2.tampilkanSaldoDiatas500000();
42             break;
43
44         case 4:
45             list.sortingName();
46             list.tampil();
47             break;
48
49         case 5:
50             System.exit(0);
51             break;
52     }
53 }
54 }

```

Berikut adalah output dari kode program diatas

```
Pilih (1-5): 1
No Rekening  Nama      Nama Ibu  Phone      Email
16030927 3084 Wallace Chase Castro 1-458-264-3263 ligula.Nullam@tacetisociosqu.edu
16100617 0573 Darius Julian Daniel 1-357-843-0547 nec@lectusjusto.org
16240401 2243 Fuller Finn Dunlap 571-70623 convallis@vestibulumanteipsum.org
16270525 0112 Malcolm Keane Floyd 623-0234 porttitor.tellus.non@Curabitur.ca
16971204 2416 Geoffrey Stephen Pratt 1-683-416-8323 ut.pellentesque@luctusutpellentesque.com
16100727 8862 Rudyard Charles Morales 650-5379 Proin.eget@velitegestaslacinia.ca
16460329 4259 Troy Damon Guerra 897-7608 pede.suspendisse.dui@a.ca
16320421 3437 Alec Cooper Lee 792-4447 non@mus.com
16180729 7229 Walter Seth Drake 863-8209 Pellentesque.ut.ipsu@neque.ca
16950313 6823 Simon Burton Gates 592-6919 tellus.justo.sit@commodoauactor.net
16850708 3528 Kamal Odysseus Salas 1-115-339-7678 dictum@nec.edu
16080205 9329 Xenos Colin Curry 1-891-703-2664 vel@ullamcorper magna.co.uk
16080628 2695 Merritt Clarke Roman 1-978-632-5110 Aliquam.gravida@vestibulumMauris.net
16130909 2979 Ryder Joel Cunningham 817-1766 interdum.Curabitur.dictum@rutrumurna.edu
16890212 8688 Preston Brock Schroeder 1-675-400-4501 et.ultrices@a.co.uk
16141008 9963 Alec Baker Barton 527-9085 ut@aultriciesadipiscing.ca
16511222 7763 Price Ashton Burke 1-564-419-4285 Proin.velit@Duisac.net
16720623 0943 Devin Slade Flores 977-6690 ac@nibhAliquamornare.com
16771126 7372 Ignatius Lionel Kane 353-5137 cubilia.Curae.Phasellus@Duis.com
```

-.:UTS:-

1. Tampil data rekening
2. Tampil data transaksi
3. Mencari saldo > 500000
4. Sorting by name
5. Keluar

Pilih (1-5): 2

No Rekening	saldo	saldoAwal	saldoAkhir	tanggalTransaksi
16030927 3084	898214.0	494048.0	3587.0	2021-03-09 07:54:42
16100617 0573	205420.0	200162.0	775880.0	2021-06-25 10:23:00
16240401 2243	838632.0	350929.0	328316.0	2021-09-18 23:00:04
16270525 0112	230659.0	204434.0	690503.0	2022-02-02 19:10:34
16971204 2416	770592.0	334245.0	444267.0	2020-08-11 13:36:56
16100727 8862	685302.0	451002.0	376442.0	2020-05-23 07:34:53
16460329 4259	816129.0	851403.0	597842.0	2021-07-18 19:41:20
16320421 3437	989609.0	333823.0	802752.0	2022-02-01 01:13:11
16180729 7229	297103.0	396116.0	779589.0	2021-07-03 01:09:49
16950313 6823	66190.0	259150.0	619774.0	2021-09-09 03:57:30
16850708 3528	234301.0	278309.0	547922.0	2021-08-24 13:18:39
16080205 9329	243306.0	869468.0	50283.0	2021-03-12 03:40:16
16080628 2695	371045.0	991242.0	602034.0	2021-08-06 11:48:59
16130909 2979	395170.0	97058.0	472273.0	2021-05-02 10:53:31
16890212 8688	862731.0	561908.0	109431.0	2021-07-31 08:11:00
16141008 9963	556798.0	31387.0	725426.0	2021-03-27 06:18:20
16511222 7763	873982.0	896213.0	846142.0	2021-07-18 04:06:30
16720623 0943	774247.0	739406.0	775848.0	2020-10-24 01:39:00
16771126 7372	66987.0	823014.0	868772.0	2020-12-21 05:57:59

-.:UTS:-

1. Tampil data rekening
2. Tampil data transaksi
3. Mencari saldo > 500000
4. Sorting by name
5. Keluar

Pilih (1-5): 3

Daftar No Rekening dengan Saldo di atas 500000:

```
-----  
16100617 0573 775880.00  
16270525 0112 690503.00  
16460329 4259 597842.00  
16320421 3437 802752.00  
16180729 7229 779589.00  
16950313 6823 619774.00  
16850708 3528 547922.00  
16080628 2695 602034.00  
16141008 9963 725426.00  
16511222 7763 846142.00  
16720623 0943 775848.00  
16771126 7372 868772.00
```

Pilih (1-5): 4				
No Rekening	Nama	Nama Ibu	Phone	Email
16320421 3437	Alec	Cooper Lee	792-4447	non@mus.com
16141008 9963	Alec	Baker Barton	527-9085	ut@aultriciesadipiscing.ca
16100617 0573	Darius	Julian Daniel	1-357-843-0547	nec@lectusjusto.org
16720623 0943	Devin	Slade Flores	977-6690	ac@nibhAliquamornare.com
16240401 2243	Fuller	Finn Dunlap	571-70623	convallis@Vestibulumanteipsum.org
16971204 2416	Geoffrey	Stephen Pratt	1-683-416-8323	ut.pellentesque@luctusutpellentesque.com
16771126 7372	Ignatius	Lionel Kane	353-5137	cubilia.Curae.Phasellus@Duis.com
16850708 3528	Kamal	Odysseus Salas	1-115-339-7678	dictum@nec.edu
16270525 0112	Malcolm	Keane Floyd	623-0234	porttitor.tellus.non@Curabitur.ca
16080628 2695	Merritt	Clarke Roman	1-978-632-5110	Aliquam.gravida@vestibulumMauris.net
16890212 8688	Preston	Brock Schroeder	1-675-400-4501	et.ultrices@a.co.uk
16511222 7763	Price	Ashton Burke	1-564-419-4285	Proin.velit@Duisac.net
16100727 8862	Rudyard	Charles Morales	650-5379	Proin.eget@velitegestaslacinia.ca
16130909 2979	Ryder	Joel Cunningham	817-1766	interdum.Curabitur.dictum@rutrumurna.edu
16950313 6823	Simon	Burton Gates	592-6919	tellus.justo.sit@commodoauctor.net
16460329 4259	Troy	Damon Guerra	897-7608	pede.Suspendisse.dui@a.ca
16030927 3084	Wallace	Chase Castro	1-458-264-3263	ligula.Nullam@tacitisociosqu.edu
16180729 7229	Walter	Seth Drake	863-8209	Pellentesque.ut.ipsum@neque.ca
16080205 9329	Xenos	Colin Curry	1-891-703-2664	vel@ullamcorper magna.co.uk

-:UTS:-

1. Tampil data rekening
2. Tampil data transaksi
3. Mencari saldo > 500000
4. Sorting by name
5. Keluar

Pilih (1-5): 5

PS C:\Documents\Kuliah\Semester 2\Algoritma Struktur Data\03\_Alfín\_ASD> █