

Bir Ulaşım Ağının Topolojik Olarak Bilgisayarda Tasarlanması, Bu Topolojide Herhangi İki Düğüm Arasından Geçen Otobüs İsimlerinin Belirtilmesi.

A. Coşkun SÖNMEZ *

Bu yazıda amaç kısaca şöyledir. Herhangi bir durakta bulunduğu-muzu farzedelim. Ve bu duraktan başka bir yere (durağa) gitmek istiyoruz. Durakta bulunan bir terminalden örneğin çağrı düğmesine basıyoruz, sonra gitmek istediğimiz durağın kot numarasını (veya doğrudan durağın adını) veriyoruz. Bu işlemin sonunda çıkışta aranılan yere giden otobüslerin isimlerini elde ediyoruz.

Eurada gaye göstergeli bir sistemi ortaya koymaktır. Örneğin merkezde bir bilgisayar, bu bilgisayara özel hat veya telefon hattıyla bağlı terminallerden oluşmuş sistemi alabiliriz. Terminal biçiminin nasıl olacağı donanım olarak tasarlanıp yapılabilir. Bu terminallerin çıkışı yazılı olabileceği gibi tablo (harita) gösterimli biçimde olabilir. Giriş ise direkt daktilo yazılımlı olabileceği gibi sayısal kotlanmış girişi sağlayacak biçimde 0-9 sayısal tuşlu ve bir kaç fonksiyon tuşlu olabilir. Bunun basit bir örneği bankamatik'dir.

Ulaşım sisteminin topolojik yapısını ileride anlatılacağı biçimde elimizde bulunan makinada temsil edildikten sonra, sonuçlarda görüleceği üzere girişi, direkt sistem girişine data kartları ile verildi. Verilen data kartlarında, çağrı yapan durağın ve aranılan durağın ismi belirtilmiştir.

Benzer biçimde bu yöntem kompleks bir hizmet birimine de uygulanabilir. Bu birimler; hastahane, kampüs, süpermarket gibi büyük ve kalabalık yerler olabilir. Örneğin; birhastahane bina katlarının belli yerlerine konulacak bu tür terminalleri bir sistemle, çıkış kapısı, ameliyathane, hasta odaları, servis birimleri öğrenilebilir. Sonuçta yine cevabı terminalden yazılı veya bir tablodan gösterimli olarak alabiliriz.

Not : Bu çalışmamın hazırlanmasında değerli yardımlarından dolayı sayın, Prof. Dr. Nadir YÜCEL'e teşekkürlerimi bir borç bilirim.

* Asıstan, S.D.M.M. Akademisi.

PROBLEMİN İNCELENMESİ

Başlangıçta bir ulaşım sistemini makinada topolojik olarak nasıl temsil edebiliriz? Topolojik yapıda düğümler ve bu düğümlerin hanelerinin birbiri ile bağlı olduklarının belirlenmesi önemli olacaktır. Ayrıca düğüm bağlantılarını sağlayan yollar yönlü ise işler daha da zorlaşacaktır.

Problemde gerçeğe yakın olarak İstanbul'un çok işlek olan duraklarını ve bu duraklardan işleyen otobüsleri alarak bir topolojik graf oluşturdum. Bu grafda düğümler duraklara karşılık, ayrıtlar ise otobüslerin işlediği yollara karşılık düşürülmüştür. Buna göre her durak için bu duraktan geçen otobüslerin belirtildiği küçük liste yapılarından oluşmuş bir dizi oluşturdum.

Liste yapısını ileride de değineceğimiz gibi makinada iki ayrı dizi biçiminde belirtiyoruz. Bu durumda ana dizimiz aslında iki diziden oluşmaktadır. 1. dizi verilerin bulunduğu dizi, 2. dizi ise veri bağlantılarını sağlayan indislerin bulunduğu dizidir.

Burada dikkate alınan, topolojik graf üzerinde, düğümler arasında bağlantıyı sağlayan yollar üzerinde işleyen otobüs seferleridir. İki düğüm arasında işleyen bir otobüs, bu yol üzerinde bulunan tüm düğümlere uğrayacaktır. Dolayısıyla bu durakların otobüs listelerinde sözü edilen otobüsün kod numarası bulunacaktır.

İşin ayrıntısına inilirse bazı sorunlar ortaya çıkacaktır. Fakat bu sorunlar çözülebilir. Örneğin; çağrı gelen durak yolun hangi tarafındadır. Yolun sağında mı solunda mı. Buna göre aranılan durağa gidiş çağrı yapılan durakla ters istikamette bulunabilir. Bunun bir çözümü ana dizideki durak listelerindeki otobüs kodlarının yolun sağ ve soluna göre (+) veya (-) işareti konmasıdır. Yani yolun sol tarafında bulunan tüm durakların otobüs kod numarası (+) dir, sağ tarafı (-) dir. Diğer bir problem ring seferlerde olacaktır. Ring seferinde kısa bir mesafeye gitmek için tüm şehri dolaşmak söz konusu olacaktır. Problem ring seferleri yapan otobüslerin hangi yöne gittikleri şimdilik öncelikli yazılış biçimine göre düşünülerek halledilmeye çalışılmıştır.

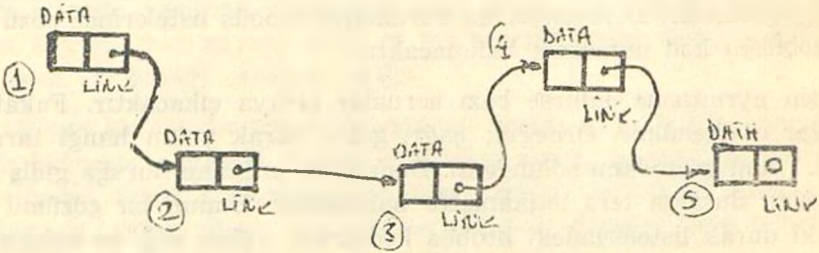
Sistem nasıl çalışacak? Bunu örnekle açıklayalım. Çağrı yapan durak ile aranılan durak listesine bakılacak. Bu iki durak (düğüm) arasından geçen ortak otobüs varsa, durak listelerinde ortak olarak bu otobüsün kod numarası bulunacaktır. Bu otobüsler bulunup çıkarılır. Eğer otobüs yoksa, çağrı gelen durağın en yakın aktarma yapabileceği durak

ile bağlantı yapılır. İlk önce aktarma yapılacak durak ile çağrı yapılan durak arasındaki ortak otobüsler belirlenir. Ondan sonra aktarma yapılan durak ile aranan durak arasında geçen otobüsler belirlenir. Burada aktarma yapılacak durağı belirleyen ve her durak için belirtilmiş olan yardımcı bir dizi oluşturmak gerekecektir.

Sözü edilen dizilerin programda geçen isimleri algoritma bölümünün sonunda belirtilecektir.

LİSTE YAPISI

Bilgisayarda kullanılan bir data yapısı türüdür. Bu yapıyı basit olarak şöyle açıklayabiliriz. Diyelimki elimizde iki kısımdan oluşmuş vagonlar olsun. Bu vagonların 1. kısmında bilgilerimiz olacak, 2. kısımda ise bundan sonra hangi vagonun geleceğini belirtecek adres bilgisi olacaktır. Böylece bir bilgi katarı oluşacaktır. En son vagonun başına başka vagon bulunmadığından, bu son vagonun ikinci kısmına liste sonu olduğunu belirten işaret konur. Örneğin; (0), (+) ve (-) gibi işaretlerinden biri konulabilir. Şekil 1 de görüleceği gibi listenin bir başlangıcı vardır. Bu başlangıç belirtilmelidir.

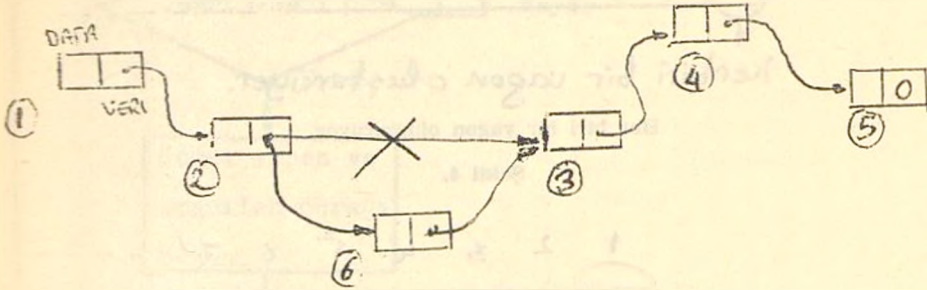


Şekil 1.

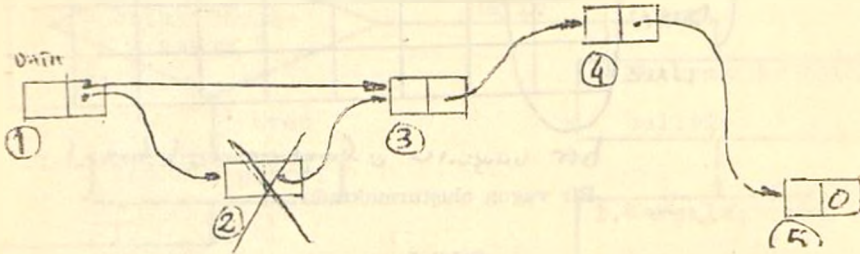
Liste yapısında, vagon örneğinde olduğu gibi yeni vagon eklenebilir. Veya çıkarılabilir. Ekleme yapılacağı zaman şekil 2'de görüleceği gibi araya yerleştirilebilir. Bu yerleştirmeye göre bağlantı yapılan bölgede LINK'de (adreslemede) gerekli bağlantı değişikliği yapılır. Şekil 2'de 6. vagonun listeye, arada bir yere yerleştirilmesi gösterilmiştir.

Şekil 3'de arada bir yerden 2. vagonun çıkarılışı gösterilmiştir. Burada yine gerekli düzeltme, değişiklik bölgesindeki LINK'de (adreslemede) yapılacaktır.

Liste yapılarını makinada nasıl tanımlayabiliriz? Basit bir liste yapısını oluşturabilmek için 2 diziye ihtiyacımız olacak. 1. dizimiz veri dizi, 2. dizimiz LİNK (adres) dizimiz olacaktır. Eğer DATA için bir di-



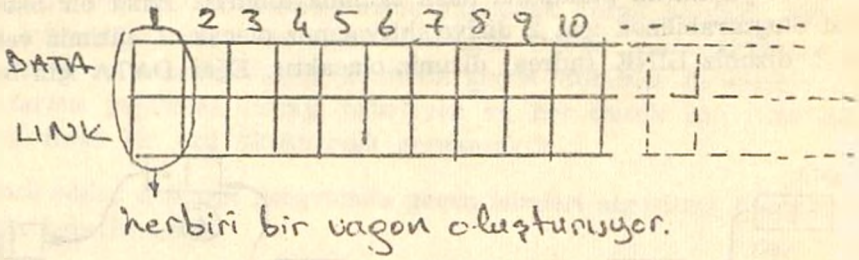
Şekil 2.



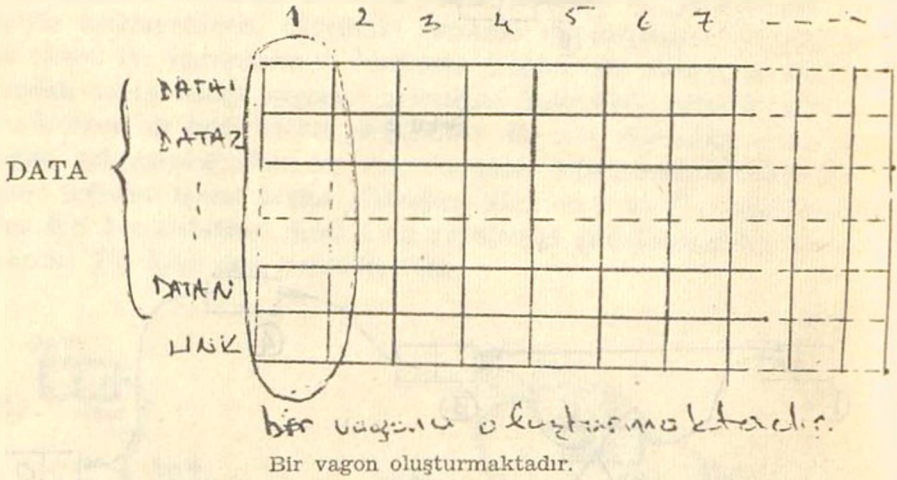
Şekil 3.

zi yetmiyorsa gerektiği kadar dizi oluşturulur. Ama yine LİNK dizimiz bir tane olacaktır. Şekil 4'de basit şekilde oluşturulmuş liste yapısını göstermektedir. Şekil 5'de ise DATA büyüklüğüne göre oluşturulmuş liste yapısını göstermektedir.

Bizim oluşturduğumuz listede OTO dediğimiz 1. dizi otobüs kod numaralarını, BAG dediğimiz 2. dizi ise LİNK (adres) indislerini içermektedir. Şekil 6'da Aksaray ve Eminönü durakları için oluşturulan liste gösterilmektedir. Burada DATA dizisinde otobüs kod numaraları bulunmaktadır.



Şekil 4.



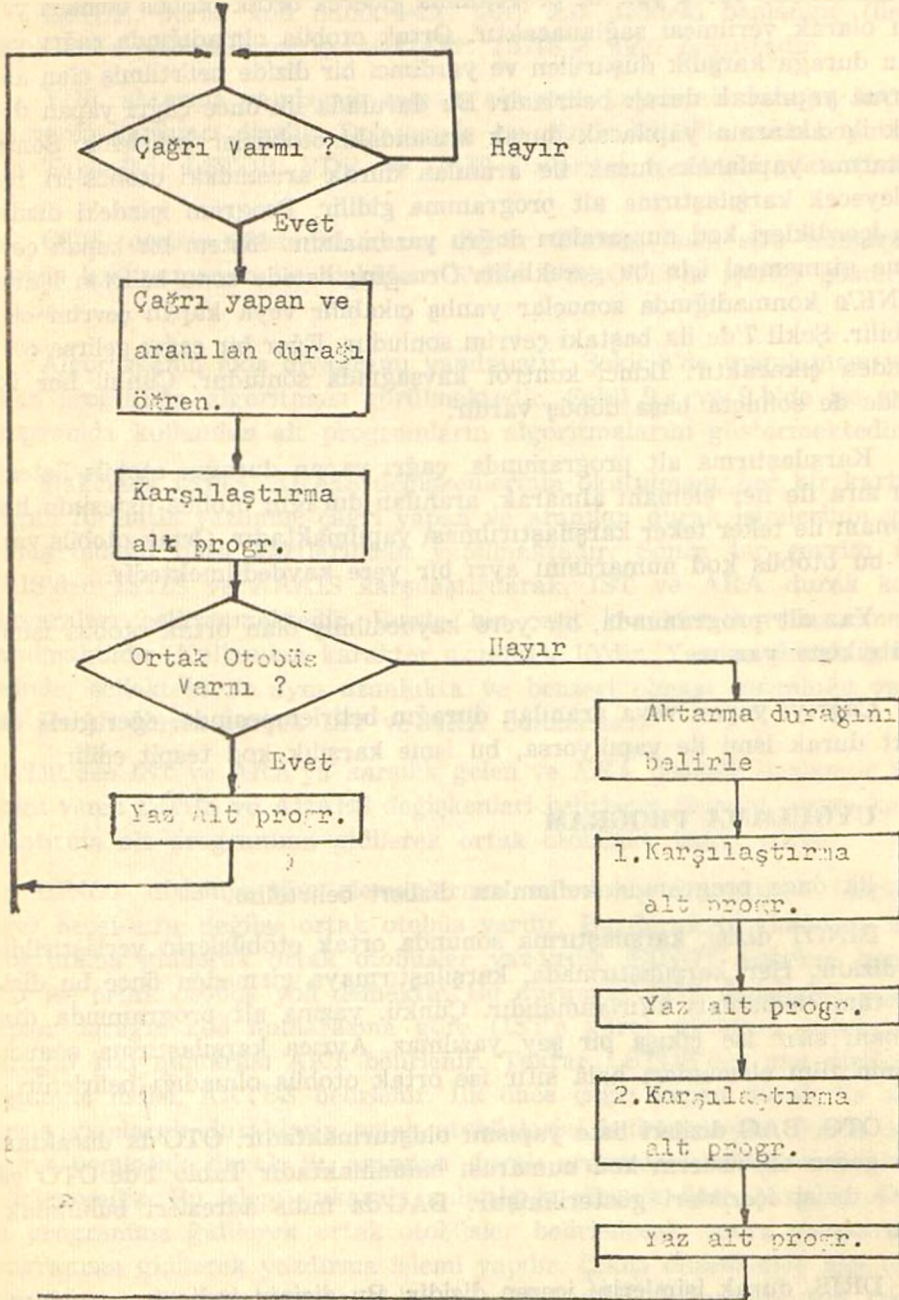
Şekil 5.

OTO	1	2	14	15	16	17	18	1	5	7	9	10	13	14	15	17	18
BAG	2	3	4	5	6	7	0	15	16	17	18	19	20	21	22	23	0

Şekil 6.

ALGORİTMA

Problemimiz, makinada oluşturduğumuz ulaşım ağından yararlanarak iki düğüm arasından geçen ortak otobüslerin belirlenmesiydi. Ohalde şekil 7'deki gösterilen algoritmada olduğu gibi ilk önce makina çağrının hangi duraktan geldiğini anlayacak. Bu çağrıyı belirledikten sonra çağrı yapan duraktan aranan durağı öğrenecektir. Bundan sonra bir karşılaştırma alt programı ile ortak otobüsleri belirleyecektir. Eğer



Şekil 7.

ortak otobüs varsa yaz alt programına giderek ortak otobüs isimleri yazılı olarak verilmesi sağlanacaktır. Ortak otobüs olmadığına çağrı yapan durağa karşılık düşürülen ve yardımcı bir dizide belirtilmiş olan aktarma yapılacak durak belirlenir. Bu durumda ilk önce çağrı yapan durak ile aktarma yapılacak durak arasındaki otobüsler belirlenir. Sonra aktarma yapılacak durak ile aranan durak arasındaki otobüsleri belirleyecek karşılaştırma alt programına gidilir. Program içindeki dizilerin içerdikleri kod numaraları doğru yazılmalıdır. Sistem bir kapalı çevrime girmemesi için bu gereklidir. Örneğin; listede sonu belirten işaret LINK'e konmadığında sonuçlar yanlış çıkabilir veya kapalı çevrim oluşabilir. Şekil 7'de ilk baştaki çevrim sonludur. Eğer bir çağrı gelirse çevrimden çıkacaktır. İkinci kontrol kavşağında sonludur. Çünkü her iki halde de sonuçta başa dönüş vardır.

Karşılaştırma alt programında, çağrı yapan durağın otobüs listesinin sıra ile her elemanı alınarak, aranan durağın otobüs listesinin her elemanı ile teker teker karşılaştırılması yapılmaktadır. Ortak otobüs varsa, bu otobüs kod numarasını ayrı bir yere kaydedilmektedir.

Yaz alt programında, bir yere kaydedilmiş olan ortak otobüs isimleri çıkışta yazılır.

Çağrıyı yapan veya aranan durağın belirlenmesinde, eğer giriş direkt durak ismi ile yapıyorsa, bu isme karşılık kod tespit edilir.

UYGULAMA PROGRAM

İlk önce programda kullanılan dizileri belirtelim.

BİNOT dizisi, karşılaştırma sonunda ortak otobüslerin yerleştirildiği dizidir. Her karşılaştırmada, karşılaştırmaya girmeden önce bu dizinin tüm elemanları sıfırlanmalıdır. Çünkü, yazma alt programında dizi elemanı sıfır ise çıkışa bir şey yazılmaz. Ayrıca karşılaştırma sonucu dizinin tüm elemanları halâ sıfır ise ortak otobüs olmadığı belirlenir.

OTO, BAG dizileri liste yapısını oluşturmaktadır. OTO'da duraklardan geçen otobüslerin kod numarası bulunmaktadır Tablo 1'de OTO ve BAG dizisi içerikleri gösterilmiştir. BAG'da indis adresleri bulunmaktadır.

DRIS, durak isimlerini içeren dizidir. Bu dizinin indis sıra numarası aynı zamanda durakların kod numarasına uygundur.

LSTDR, durak kod numarasına göre ana dizideki başlangıç (liste başı) indis numaralarını içermektedir. DRIS'le aynı boyuttur.

YDR, aktarma yapılacağı zaman, aktarma yapılacak durağın kodunu veren yardımcı dizidir. Dolayısıyla bu dizide DRIS'le aynı boyuttur. Tablo 2'de LSTDR, YDR, ve DRIS dizilerinin içerikleri gösterilmiştir.

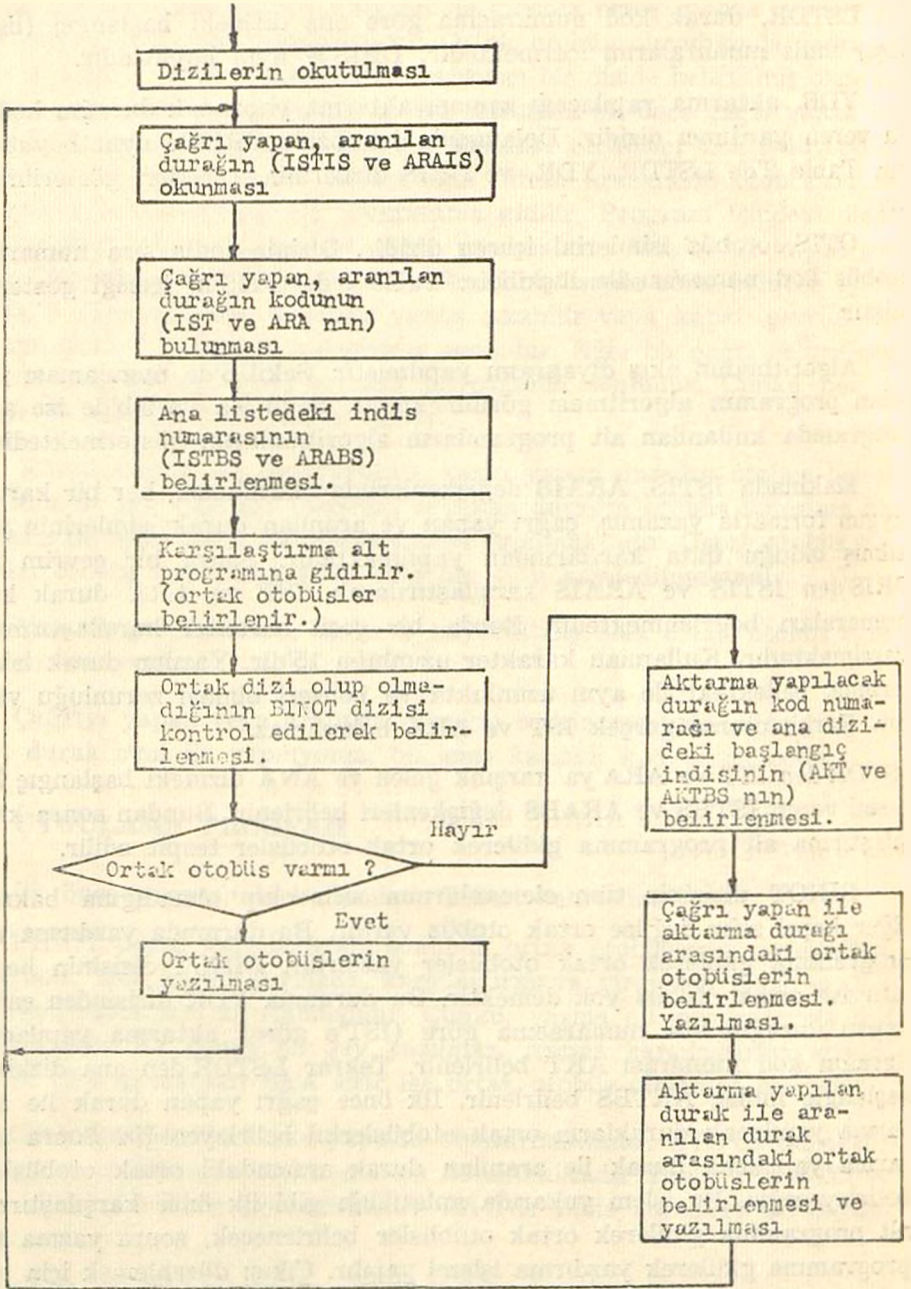
OTIS, otobüs isimlerini içeren dizidir. Dizinin indis sıra numarası otobüs kod numarası ile ilişkilidir. Tablo 3'de OTIS'in içeriği gösterilmiştir.

Algoritmanın akış diyagramı yapılmıştır. Şekil 8'de uygulaması yapılan programın algoritması görülmektedir. Şekil 9.a ve 9.b'de ise ana programda kullanılan alt programların algoritmalarını göstermektedir.

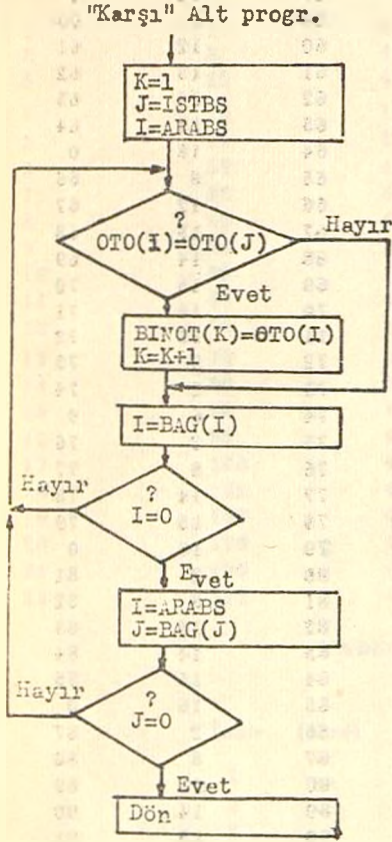
Makinada ISTIS, ARAIS değişkenlerinin okutulması, her bir kartta uygun formatla yazılmış, çağrı yapan ve aranılan durak isimlerinin yazılmış olduğu data kartlarından yapılmaktadır. Sonra bir çevrim ile DRIS'den ISTIS ve ARAIS karşılaştırılarak, IST ve ARA durak kod numaraları belirlenmektedir. Burda bir çeşit karakter karşılaştırması yapılmaktadır. Kullanılan karakter uzunluğu 15'dir. Yazılan durak isimlerinde, bellekteki ile aynı uzunlukta ve benzeri olması zorunluğu vardır. Farklı olursa gerçek IST ve ARA bulunamaz.

ISTDR'den IST ve ARA'ya karşılık gelen ve ANA dizideki başlangıç indisini veren ISTBS ve ARABS değişkenleri belirlenir. Bundan sonra karşılaştırma alt programına gidilerek ortak otobüsler tespit edilir.

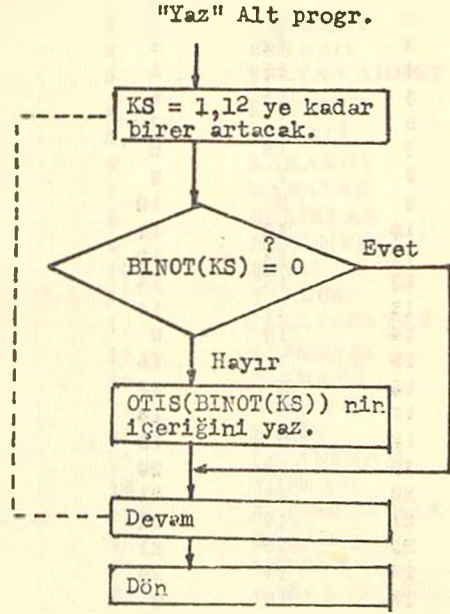
BINOT dizisinin tüm elemanlarının sıfır olup olmadığına bakılır. Eğer hepsi sıfır değilse ortak otobüs vardır. Bu durumda yazdırma alt programına gidilerek ortak otobüsler yazdırılır. BINOT dizisinin hepsi sıfır ise ortak otobüs yok demektir. Bu durumda YDR dizisinden çağrı yapan durağın kod numarasına göre (IST'e göre) aktarma yapılacak durağın kod numarası AKT belirlenir. Tekrar LSTDR'den ana dizideki başlangıç indisi, AKTBS belirlenir. İlk önce çağrı yapan durak ile aktarma yapılacak durakların ortak otobüslerini belirleyeceğiz. Sonra aktarma yapılacak durak ile aranılan durak arasındaki ortak otobüsleri belirleyeceğiz. Bu işlem yukarıda anlatıldığı gibi ilk önce karşılaştırma alt programına gidilerek ortak otobüsler belirlenecek, sonra yazma alt programına gidilerek yazdırma işlemi yapılır. Çıkışı düzenlemek için uygun format düzenlerinde başlıklar yapılmaktadır. Bunu program çıktısında görmekteyiz.



Şekil 8.



Şekil 9a.



Şekil 9b.

S O N U Ç

Bir ulaşım ağının makinada temsil edildiği bu yazının projesinde, ulaşım ağının herhangi iki noktası arasındaki ortak otobüslerin belirlenmesinde görüldüğü gibi liste yapısından faydalanılmıştır. Liste yapısı özelliğinden faydalanarak ulaşım ağına sonradan yeni durak ve otobüs seferleri eklenebilir. Ayrıca küçük alt ulaşım ağları oluşturarak, bunlar arasında ilişki kurulabilir. Örneğin; İstanbul'un Anadolu, Avrupa yakası gibi iki alt ulaşım ağı oluşturabiliriz.

İndis	OTO	BAG	İndis	OTO	BAG
1	1	2	58	18	0
2	2	3	59	3	60
3	14	4	60	12	61
4	15	5	61	13	62
5	16	6	62	14	63
6	17	7	63	15	64
7	18	0	64	18	0
8	1	9	65	8	66
9	9	10	66	12	67
10	10	11	67	13	68
11	14	12	68	14	69
12	15	13	69	15	70
13	17	14	70	16	71
14	18	0	71	18	72
15	1	16	72	3	73
16	5	17	73	2	74
17	7	18	74	6	0
18	9	19	75	2	76
19	10	20	76	8	77
20	13	21	77	14	78
21	14	22	78	15	79
22	15	23	79	16	0
23	17	24	80	2	81
24	18	0	81	8	82
25	1	26	82	11	83
26	9	27	83	14	84
27	10	28	84	15	85
28	14	29	85	16	0
29	15	30	86	2	87
30	17	31	87	8	88
31	18	0	88	11	89
32	5	33	89	14	90
33	7	34	90	15	91
34	11	35	91	16	92
35	13	36	92	10	93
36	14	37	93	11	94
37	15	38	94	2	95
38	18	0	95	9	96
39	4	40	96	8	97
40	5	41	97	14	98
41	6	42	98	15	99
42	7	43	99	16	0
43	11	44	100	8	101
44	13	45	101	9	0
45	14	46	102	1	103
46	15	47	103	2	104
47	18	0	104	16	105
48	4	49	105	17	106
49	5	50	106	18	0
50	6	51	107	1	108
51	7	52	108	2	0
52	11	0	109	3	110
53	3	1	110	4	111
54	12	2	111	5	0
55	13	3	112	6	113
56	14	4	113	7	0
57	15	5	114		
		8	115		

TABLO 1

İndis	LSTDR	YDR	Kod	DRİS
1	1	4	1	AKSARAY
2	8	4	2	BEYAZIT
3	25	4	3	SULTAN AHMET
4	15	11	4	EMİNÖNÜ
5	15	11	5	SİRKECİ
6	32	4	6	KARAKÖY
7	39	4	7	KABATAŞ
8	48	4	8	BEŞİKTAŞ
9	53	11	9	MECİDİYEKÖY
10	59	11	10	ŞİŞLİ
11	65	4	11	TAKSİM
12	75	11	12	GALATASARAY
13	75	11	13	TEPEBAŞI
14	80	11	14	ŞİŞHANE
15	86	11	15	UNKAPANI
16	92	4	16	FATİH
17	100	4	17	EDİRNEKAPI
18	102	4	18	TOPKAPI
19	107	4	19	BAKIRKÖY TARAFI
20	109	11	20	BOĞAZ
21	109	4	21	LEVENT
22	112	4	22	BOĞAZ TARAFI

TABLO 2

İndis	(Kod)	OTIS (Otobüs isimleri)
1		EMİNÖNÜ BAKIRKÖY
2		TAKSİM BAKIRKÖY
3		TAKSİM LEVT-BOĞZ
4		TAKS. BEŞT-BOĞZ
5		EMİNÖNÜ LEVT.-BOĞZ
6		TAKSİM BOĞAZ
7		EMİNÖNÜ BOĞAZ
8		TAKSİM EDİRNEKAPI
9		EMİNÖNÜ EDİRNEKAPI
10		EMİNÖNÜ FATİH
11		BEŞİKTAŞ FATİH
12		TAKSİM MECİDİYEKÖY
13		EMİNÖNÜ MECİDİYEKÖY
14		TAKSİM BEYAZIT-A
15		TAKSİM BEYAZIT-B
16		TAKSİM TOPKAPI
17		EMİNÖNÜ TOPKAPI
18		MECİDİYEKÖY TOPKAPI

TABLO 3

CAGRIYI YAPAN DURAK :

AKSARAY

ARANILAN DURAK :

BESIKTAS

AKTARNALI GIDECEK SINIZ

AKSARAY DEN EMIRONU

E GIDEN OTOBUSLER

EMINONU SAKIRKOY

TAKSIM BEYAZIT A

TAKSIM BEYAZIT B

EMINONU TOPKAPI

HECIDIYEKÖY TOPKAPI

EMINONU DE AKTARNA YAPACAKSINIZ. CRADA ASAGIDAKI OTOBUSLERE BINEBILIRSINIZ.

EMINONU

DEN

BESIKTAS

E GIDEN OTOBUSLER

EMINONU LEVT. BOGZ.

EMINONU BOGAZ