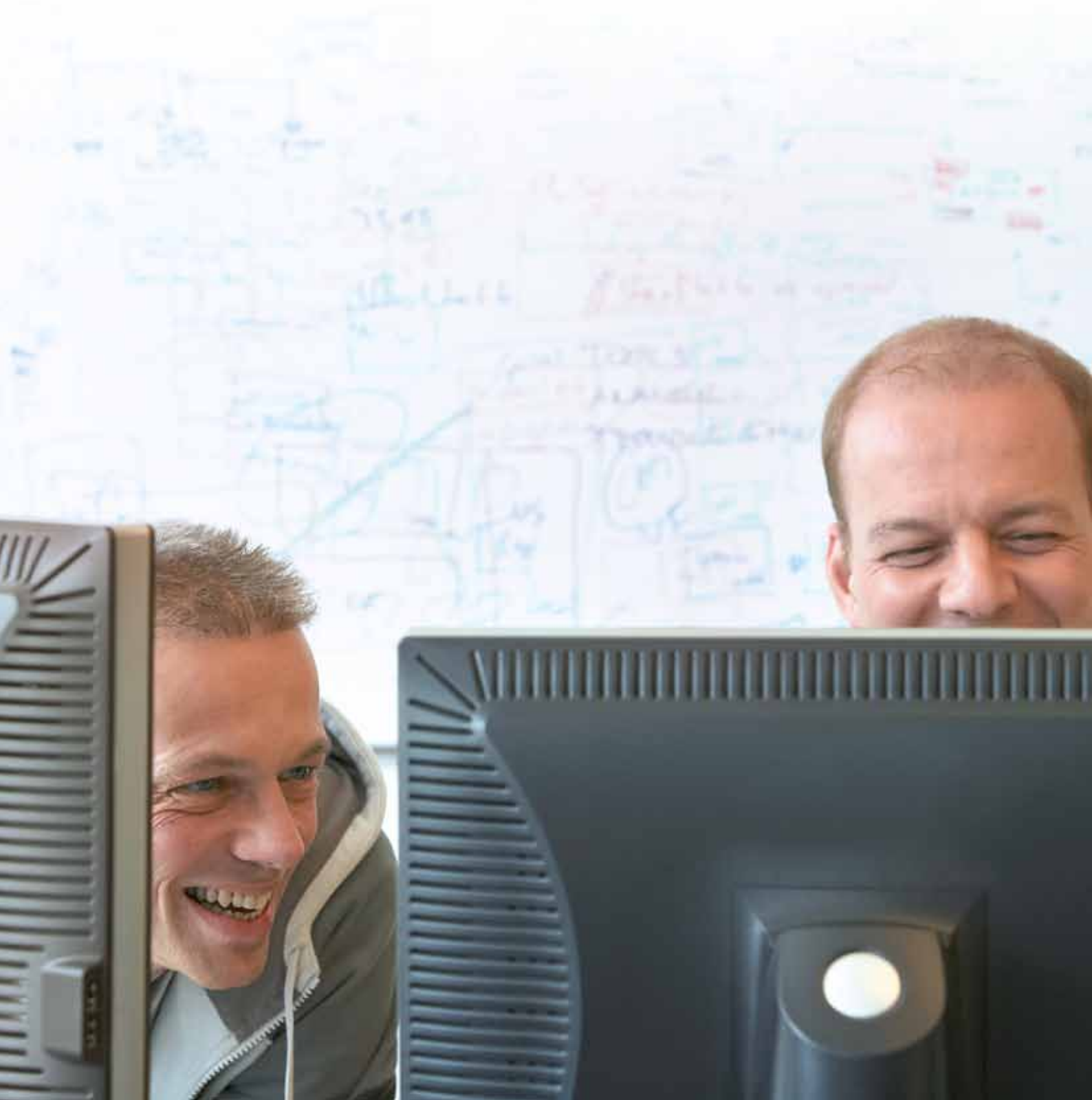


ADINOVUM

ENGINEERING THE IMPOSSIBLE.



```

5.     public E edge;
6.     public DijkstraInfo()
7.     {
8.         this.distance = Integer.MAX_VALUE;
9.         this.predecessor = null;
10.        this.edge = null;
11.    }
12.    public String toString()
13.    {
14.        return "["+this.distance+","+this.predecessor+","+this.edge+"]";
15.    }
16. }
17. private static class DijkstraComp<V extends Vertex, E extends WeightedEdge<V>>
18.     implements Comparator<V>
19.     {
20.         private HashMap<V, DijkstraInfo<E>> info;
21.         public DijkstraComp(HashMap<V,DijkstraInfo<E>> Info)
22.         {
23.             this.info = Info;
24.         }
25.         public int compare(V a, V b)
26.         {
27.             int ad = this.info.get(a).distance;
28.             int bd = this.info.get(b).distance;
29.             if(ad == bd) return 0;
30.             if(ad < bd) return -1;
31.             return 1;
32.         }
33.     }
34.     public static<V extends Vertex, E extends WeightedEdge<V>>
35.     HashMap<V,Vector<E>> compute(Graph<V,E> G, V s)
36.     {
37.         Vector<V> vertices = G.getVertices();
38.         Vector<E> edges = G.getEdges();
39.         HashMap<V, Vector<E>> adjList = G.getAdjacencyList();
40.         HashMap<V, DijkstraInfo<E>> info = new HashMap<V,DijkstraInfo<E>>();
41.         PriorityQueue<V> Q = new PriorityQueue<V>(edges.size(), new
42.         DijkstraComp<V,E>(info));
43.         for(Enumeration<V> e=vertices.elements(); e.hasMoreElements(); )
44.             info.put(e.nextElement(), new DijkstraInfo<E>());
45.         info.get(s).distance = 0;
46.         Q.addAll(vertices);
47.         while(!Q.isEmpty())
48.         {
49.             V v = Q.poll();
50.             for(Enumeration<E> e = adjList.get(v).elements(); e.hasMoreElements(); )
51.             {
52.                 E t = e.nextElement();
53.                 if(info.get(v).distance + t.getWeight() < info.get(t.getEndVertex()).distance)
54.                 {
55.                     info.get(t.getEndVertex()).distance = info.get(v).distance + t.getWeight();
56.                     info.get(t.getEndVertex()).predecessor = v;
57.                     info.get(t.getEndVertex()).edge = t;
58.                     //Q erneuern
59.                     Q.remove(t.getEndVertex());
60.                     Q.add(t.getEndVertex());

```



LEISTUNGEN END-TO-END UND DARÜBER HINAUS.

AdNovum entwickelt Applikationen, Sicherheitssoftware und Integrationslösungen massgeschneidert für den Einsatz im High-End-Bereich. Als renommiertes Softwarehaus bietet AdNovum ihren Kunden nebst der Softwareentwicklung umfassende Unterstützung in der Projektabwicklung von der Bedürfnisanalyse und Spezifikation bis hin zur Integration und produktiven Einführung der Soft- und Hardware.

Projektspezifisch zusammengestellte Expertenteams erarbeiten überzeugende Resultate in bester Schweizer Ingenieurtradition. Auf AdNovum ist Verlass: Ihre Markenzeichen sind technische Kompetenz und weitreichende Erfahrung in der Entwicklung anspruchsvoller Lösungen. Zu ihrem Kundenstamm zählen namhafte Unternehmen und Behörden, die mit besonders schützenswerten Daten arbeiten.

AdNovums Leistungsangebot umfasst:

Entwicklung von Geschäftsapplikationen und Webanwendungen

- E-Banking, Banking-Business
- Secure ECM (Enterprise-Content-Management)
- CRM, Contact-Center
- E-Health und E-Government
- Logistik, ERP
- Facility-Management

Security-Engineering

- Single-Signon-Portale (SSO)
- Revisionsfähige Gesamtsysteme, End-to-End-Security
- Access- und Identity-Management
- Sichere Kommunikation

Integration, Migration, Software-Renovation

- Systemintegration von Mainframes, Servern, Workstations, Legacy-Systemen, Datenbanken, Telefonie
- Integration von Drittsystemen (Kernbankensysteme, SAP-Certified Integration)
- Schnittstellenlösungen, Middlewarekomponenten, Protokoll- und Datenkonversion
- Daten- und Technologiemigration
- Renovation von Legacy-Systemen

IT-Consulting

- Security- und Technologie-Reviews
- IT-Architektur, IT-Security, IT-Strategie

Wartung, Support

- 7 x 24-h-Support, 2nd-/3rd-Level-Applikationssupport
- Wartung und Pflege von Legacy-Applikationen



```
1. import std.math;
2. public class MergeSort
3. {
4.     public void sort(int[] Array)
5.     {
6.         this.mergeSort(Array,0,Array.length-1);
7.     }
8.     private void mergeSort(int[] A, int f, int l)
9.     {
10.        if(f < l)
11.        {
12.            int m = cast(int)floor((l+f)/2);
13.            this.mergeSort(A, f, m);
14.            this.mergeSort(A, m+1, l);
15.            this.merge(A, f, m, l);
16.        }
17.    }
18.    private void merge(int[] A, int f, int m, int l)
19.    {
20.        int tmp[] = new int[(l-f)+1];
21.        int ai = f;
22.        int bi = m+1;
23.        for(int i=0; i < tmp.length;i++)
24.        {
25.            if(ai <= m && (bi > l || (A[ai]< A[bi])) )
26.            {
27.                tmp[i] = A[ai];
28.                ai++;
29.            }
30.            else
31.            {
32.                tmp[i] = A[bi];
33.                bi++;
34.            }
35.        }
36.        for(int i=0; i < tmp.length; i++)
37.            A[f+i] = tmp[i];
38.    }
39. }
40. }
41. }
42. import std.stdio;
43. public void main()
44. {
45.     int[] A = [4,30,1,2,3,35,37,88,99];
46.     MergeSort algo = new MergeSort();
47.     algo.sort(A);
48.     foreach(int e; A)
49.         writef(e, " ");
50. }
```



Lösungen

PRAXISNAH, ABER FERN VON NULLACHTFÜNFZEHN.

AdNovum liefert starke IT-Lösungen für anspruchsvolle Kunden.

AdNovum arbeitet lösungsorientiert:

- klarer Fokus auf Engineering und Security
- praxisnahe adaptive Projektmanagementprozesse

Applikationen von AdNovum zeichnen sich aus durch:

- höchste Zuverlässigkeit, Performance und Skalierbarkeit
- bewährte Integrations- und Sicherheitskomponenten
- konsequente Abdeckung von Marktvorgaben, gesetzlichen und regulatorischen Auflagen
- Einsatz offener State-of-the-Art-Technologien und -Standards

Projekte

FÜR LANGE LEBENSZYKLEN UND SHORTCUTS-TO-MARKET.

AdNovum realisiert Grossprojekte als Generalunternehmer (GU) oder im Verbund mit Partnern. Im Zentrum steht dabei immer der Kunde. Langjährige Kundenbeziehungen zeugen vom nachhaltigen Mehrwert der erbrachten Leistung.

Als Kunde wissen Sie, dass Ihr Projekt gelingt:

- Sie kennen Ihre Ansprechpartner und arbeiten mit den besten Leuten zusammen.
- Sie haben die Gewähr, dass Ihre Ressourcen unter hohem Investitionsschutz optimal eingesetzt werden.
- Sie können darauf bauen, dass Ihre neue Lösung in der gewünschten Qualität termingerecht realisiert wird.
- Sie reduzieren Ihre IT-Kosten und verkürzen die Time-to-Market.
- Sie können sich darauf verlassen, dass der gesamte Software-Lebenszyklus abgedeckt ist und Änderungen schnell und flexibel umgesetzt werden.
- Sie erhöhen mit innovativen Anwendungen die Wertschöpfung und vergrössern den Vorsprung zu Ihren Mitbewerbern.

IT-Consulting

INGENIEURE ALS BERATER.

AdNovum gibt ihre Erfahrung aus der Umsetzung anspruchsvoller IT-Projekte in Beratungsmandaten und Reviews an ihre Kunden weiter. In der Beratung setzt AdNovum Softwareingenieure ein. Als Kunde haben Sie dadurch Gewähr, dass Ihr Berater über aktuelle Erfahrung verfügt und jederzeit auf den Know-how-Pool der AdNovum zugreifen kann.

Wartung und Support

FULL SERVICE AUS EINER HAND.

AdNovum bietet umfassende Wartungs- und Supportdienstleistungen. Im Dialog mit dem Kunden werden schlanke Service-Level-Agreements (SLAs) erarbeitet, die genauso individuell und an die Kundenbedürfnisse angepasst sind wie die IT-Lösungen. AdNovum übernimmt auch Wartung und Support von Systemen, die von Kunden oder Drittfirmen realisiert wurden.

```

2. import java.util.Enumeration;
3. import java.util.PriorityQueue;
4. import java.util.Vector;
5. public class Kruskal
6. {
7.     public static <V extends Vertex,E extends WeightedEdge<V>> Vector<E> compute (UndirectedGraph<V,E> G)
8.     {
9.         Vector<V> vertices = G.getVertices();
10.        Vector<E> edges = G.getEdges();
11.        UnionFind uf = new UnionFind();
12.        Vector<E> mST = new Vector<E>();
13.        for(Enumeration<V> e = vertices.elements(); e.hasMoreElements(); )
14.        {
15.            uf.createUnion(e.nextElement());
16.        }
17.        PriorityQueue<E> Q = new PriorityQueue<E>(edges.size(),new Comparator<E>(){
18.            public int compare(E a, E b)
19.            {
20.                int aw = a.getWeight();
21.                int bw = b.getWeight();
22.                if (aw==bw)return 0;
23.                if (aw<bw)return -1;
24.                return 1;
25.            }
26.        });
27.        Q.addAll(edges);
28.        while(Q.size() > 0)
29.        {
30.            E e = Q.poll();
31.            if(uf.find(e.getStartVertex()) != uf.find(e.getEndVertex()))
32.            {
33.                uf.merge(e.getStartVertex(),e.getEndVertex());
34.                mST.add(e);
35.            }
36.        }
37.        return mST;
38.    }
39. }
40. class KruskalProg
41. {
42.     public static void main(String[] Args)
43.     {
44.         Vector<Vertex> v = new Vector<Vertex>();
45.         v.add(new Vertex());
46.         v.add(new Vertex());
47.         v.add(new Vertex());
48.         Vector<WeightedEdge<Vertex>> e = new Vector<WeightedEdge<Vertex>>();
49.         e.add(new WeightedEdge<Vertex>(v.get(0),v.get(1),10));
50.         e.add(new WeightedEdge<Vertex>(v.get(1),v.get(2),15));
51.         e.add(new WeightedEdge<Vertex>(v.get(0),v.get(2),7));
52.         UndirectedGraph<Vertex, WeightedEdge<Vertex>> g = new UndirectedGraph<Vertex,
53. WeightedEdge<Vertex>>(v,e);
54.         System.out.println(g);
55.         System.out.println(Kruskal.compute(g));
56.     }
57. }
58. }

```



MEILENSTEINE FÜR MORGEN.

Als Ingenieurfirma mit Tradition setzt AdNovum auf einen Software-Engineering-Prozess, der mit der grossen Zahl realisierter Projekte gereift ist:

- klar abgegrenzte Projektphasen mit Meilensteinen und standardisierten Lieferobjekten
- effizienter und kontrollierter Einsatz innovativer Technologie dank professionellem Technologiemanagement
- Qualitätssicherungsprozess mit Prüfstellen (Architektur- und Security-Signoffs)
- automatisierte Überprüfung zentraler Qualitätsfaktoren (Sourcecode-Qualität, Kompilierbarkeit, Deploybarkeit und Lauffähigkeit)
- kontrolliertes Change-Management
- systematisches Controlling und Reporting

Technisches Umfeld

OFFENE SCHNITTSTELLEN FÜR NEUES.

AdNovum setzt bei ihren Best-of-Breed-Produkten und -Lösungen auf die gezielte Kombination von Standardprodukten bedeutender Hersteller mit Open-Source-Software und offenen Standards. Sie betreibt ein durchgängiges Technologiemanagement: Ein Expertenteam überprüft periodisch die verwendeten Technologien und sorgt für eine kontinuierliche kontrollierte Erneuerung. Das technische Umfeld ist durch folgende Eckpunkte gekennzeichnet:

- Primärplattformen Solaris und Linux
- Weitere Plattformen: Windows, Virtualisierungsplattformen
- Java/Java EE (Oracle WebLogic, IBM WebSphere, JBoss)
- Oracle-Datenbank
- Oracle UCM als ECM-System
- Eclipse als IDE
- Weitere Tools Entwicklungsumgebung: SVN, CVS, Ant, Maven, soapui, Dependency Matrix Analyzer, Junit, Grinder, JProbe
- Wichtige Frameworks: JEE5, Nevis, Spring, Hibernate, JSF, Struts, Wicket

Wissens- und Projektmanagement

GRAUE ZELLEN IN AKTION.

Die Stärken der AdNovum im Wissens- und Projektmanagement sorgen für eine nachhaltig hohe Qualität der Lieferobjekte dank:

- Kompetenz und Erfahrung ihrer Mitarbeiter
- übergreifender Planung der Projektvorhaben und Ressourcen durch speziell geschaffene Instrumente
- standardisierter und weitgehend automatisierter Projekt-Setups
- klarer Rollenzuweisung, griffiger Rollen- und Kompetenzbeschreibung, hoher Kontinuität bei der Rollenbesetzung

```
2. import java.util.HashMap;
3. import java.util.Hashtable;
4. import java.util.Vector;
5. public class BellmanFord
6. {
7.     public static<V extends Vertex, E extends WeightedEdge<V>> HashMap<V,Vector<E>> compute(Graph<V,E> G, V
8. )
9.     {
10.         Vector<V> vertices = G.getVertices();
11.         Vector<E> edges = G.getEdges();
12.         HashMap<V,Vector<E>> invAdjList = invertAdjacencyList(G.getAdjacencyList(), vertices);
13.         Hashtable<Integer, Hashtable<V,Integer>> matrix = new Hashtable<Integer,Hashtable<V,Integer>>(vertices.
14. size());
15.         for(int i=0; i < vertices.size();i++)
16.         {
17.             matrix.put(i, new Hashtable<V,Integer>(vertices.size()));
18.             for(Enumeration<V> e = vertices.elements(), e.hasMoreElements();)
19.             {
20.                 matrix.get(i).put(e.nextElement(),Integer.MAX_VALUE);
21.             }
22.             HashMap<V,E> predecessor = new HashMap<V, E>();
23.             for(Enumeration<V> e = vertices.elements(); e.hasMoreElements();)
24.                 predecessor.put(e.nextElement(), null);
25.             matrix.get(i).put(i, 0);
26.             for(int i=1; i < vertices.size();i++)
27.             {
28.                 for(Enumeration<V> ev=vertices.elements(); ev.hasMoreElements();)
29.                 {
30.                     V v = ev.nextElement();
31.                     E pre = null;
32.                     int min = matrix.get(i-1).get(v);
33.                     for(Enumeration<E> ea = invAdjList.get(v).elements(); ea.hasMoreElements();)
34.                     {
35.                         E t = ea.nextElement();
36.                         if( matrix.get(i-1).get(t.getStartVertex()) != Integer.MAX_VALUE && matrix.get(i-1).get(t.getStartVer-
37. tex()) + t.getWeight() < min )
38.                         {
39.                             min = matrix.get(i-1).get(t.getStartVertex()) + t.getWeight();
40.                             pre = t;
41.                         }
42.                     }
43.                     matrix.get(i).put(v,min);
44.                     if(pre != null)
45.                         predecessor.put(v,pre);
46.                 }
47.             }
48.             HashMap<V,Vector<E>> retval = new HashMap<V,Vector<E>>();
49.             for(Enumeration<V>e= vertices.elements(), e.hasMoreElements();)
50.                 retval.put(e.nextElement(), new Vector<E>());
51.             for(Enumeration<V>e = vertices.elements(),e.hasMoreElements();)
52.             {
53.                 Vertex v = e.nextElement();
54.                 Vertex pre = v;
55.                 while(predecessor.get(pre) != null)
```

Referenzen

LORBEEREN ALS INNOVATIONSMOTOR.

Dank ihrer Erfahrung gelingt es AdNovum, selbst bei komplexen Problemstellungen überzeugende Lösungen zu finden.

Applikationsentwicklung

- EJPD: Zentrales Migrationsinformationssystem ZEMIS, elektronisches Dossier
- UBS: Point of Sale / Schalterarbeitsplatz
- Swiss Post International: ERP-Logistiklösung GLOBE
- Kant. Steueramt Zürich: Register- und Lagerverwaltung (ReLa)
- UBS: Facility-Management (GEIS/GIKUS)
- SIX Group: PayNet, elektronische Rechnungsabwicklung (EBPP, Electronic Bill Presentment and Payment)
- BIT/BAKOM: SOA-Architektur/E-Licensing
- UBS: Enterprise-Content-Management (Secure ECM)

Security und Integration (SSO-Portale, IDM, PKI)

- UBS: Security-Stack / Zertifikatsmanagement
- EJPD: Single-Signon-Portal (SSO)
- Die Schweizerische Post: SSO-Portal «My Post Business»
- Swiss Post International: Extranet SSO-Portal
- Die Mobiliar: Web Entry Service
- PostFinance: E-Finance und pf.ch
- PostFinance: IDM/PKI und UAM (Unix-Access-Management)
- SwissSign: Postzertifikate

E-Banking und Secure-Messaging

- UBS: E-Banking via Internet und via Telefon
- Privatbank IHAG: E-Banking, Content-Management-System
- Privatbank Wegelin & Co.: E-Banking
- UBS Mailbox

IT-Consulting

- Konzeption einer sicheren Java-EE-Architektur und einer Anwendungs-Roadmap für ein Bundesdepartement
- Technische Leitung und Koordination eines Security-Programms für einen grossen Schweizer Finanzdienstleister
- Evaluation diverser Sicherheitsmittel wie Authentisierungs-Token für eine Kantonalbank
- Technische Reviews (SOA, Architektur, Code, Deployment, Performance) für diverse Kunden aus dem Finanz- und Versicherungsbereich

```
2. import java.util.Vector;
3. import java.util.HashMap;
4. import java.util.EnumSet;
5. public class BFSInfo {
6.     {
7.     private State state;
8.     {
9.     WHITE,
10.    GRAY,
11.    BLACK
12.    }
13.    static class BFSInfo
14.    {
15.    private Vertex predecessor;
16.    private Vertex vertex;
17.    private int distance;
18.    private State state;
19.    public BFSInfo(Vertex v) {
20.    this.vertex = v;
21.    this.predecessor = null;
22.    this.distance = 0;
23.    this.state = State.WHITE;
24.    }
25.    public Vertex getPredecessor()
26.    {
27.    return this.predecessor;
28.    }
29.    }
30.    public void setPredecessor(Vertex v)
31.    {
32.    this.predecessor = v;
33.    }
34.    public int getDistance()
35.    {
36.    return this.distance;
37.    }
38.    }
39.    public void setDistance(int d)
40.    {
41.    this.distance = d;
42.    }
43.    }
44.    public State getState()
45.    {
46.    return this.state;
47.    }
48.    public void setState(State s)
49.    {
50.    this.state = s;
51.    }
52.    }
53.    public static <V extends Vertex, E extends Edge<V>> HashMap<V, BFSInfo> compute(Graph<V, E> G, V start)
54.    {
55.    HashMap<V, Vector<E>> adjList = G.getAdjacencyList();
56.    Vector<V> vertices = G.getVertices();
57.    HashMap<V, BFSInfo> info = new HashMap<V, BFSInfo>(adjList.size() * 2);
58.    //Info Quellen
59.    for(Enumeration<V> e = vertices.elements(); e.hasMoreElements();)
```

SOFTWARE MIT HARTEM SCHWEIZER KERN.

Das vielseitige Projekt- und Kundenportfolio der AdNovum bietet Top-Fachleuten faszinierende und herausfordernde Aufgabenstellungen. Die Teams der AdNovum setzen sich aus hervorragend ausgebildeten Mitarbeitern zusammen, die mit viel Eigeninitiative und Technologieverständnis bestechende Lösungen erarbeiten.

Starke Firmenkultur

Ein prägendes Element der AdNovum ist ihre lösungsorientierte Engineering-Kultur, die vom Engagement und von der Loyalität ihrer Mitarbeiter getragen wird.

Ihren Mitarbeitern bietet AdNovum:

- moderne, grosszügige Arbeitsplätze
- attraktive Arbeitsbedingungen
- ein kreatives und inspirierendes Betriebsklima
- offene Kommunikationskanäle und Plattformen für den Wissensaustausch
- laufende Weiterbildung «on the job» mittels betriebsinterner und -externer Angebote bis hin zum Besuch von Nachdiplomstudiengängen

Eckdaten

Gründungsjahr: 1988

Geschäftsleitung

- Ruedi Wipf, CEO

Mitarbeiter: 200

- 70% Ingenieure (ETH/Uni/FH)

Standorte

- Zürich (HQ)
- Bern
- Singapur
- Budapest

Partnerschaften

AXSionics, the i-engineers, Oracle, SAP, SafeNet, wikima4

Hauptsitz

AdNovum Informatik AG
Ruedi Wipf, CEO
Röntgenstrasse 22, 8005 Zürich
Tel. +41 44 272 61 11, Fax +41 44 272 63 12
E-Mail: ruedi.wipf@adnovum.ch

Geschäftsstelle Bern

AdNovum Informatik AG
Adrian Schmid, Leiter Geschäftsstelle Bern
Erlachstrasse 16b, 3012 Bern
Tel. +41 31 952 58 58, Fax +41 31 952 58 60
E-Mail: info@adnovum.ch

AdNovum Singapur

AdNovum Singapore Pte. Ltd.
Martin Nokes, Managing Director
72 Anson Road, #07-01 Anson House
SG-079911 Singapore
Tel. +65 6536 0668, Fax +65 6536 2812
E-Mail info@adnovum.sg

AdNovum Ungarn

AdNovum Hungary Kft.
Chris Tanner, Managing Director
Kapás utca 11-15
H-1027 Budapest
Tel. +36 1 487 5000, Fax +36 1 487 5005
E-Mail: info@adnovum.hu

Schwestergesellschaften Zürich

AdNovum Operations AG
AdNovum Engineering AG
Röntgenstrasse 22, 8005 Zürich
Tel. +41 44 272 61 11, Fax +41 44 272 63 12
E-Mail: info@adnovum.ch

Medienkontakt

AdNovum Informatik AG
Manuel Ott, Corporate Communication
Röntgenstrasse 22, 8005 Zürich
Tel. +41 44 272 61 11, Fax +41 44 272 63 12
E-Mail: info@adnovum.ch