**מערך**

כולל התצוגה, והמלל היחיד שלכאורה נדבר, ללא הרחבות והסברים נוספים.

|  |  |
| --- | --- |
|  | שלום, אני רותי, אני פסי. אנו גאות להציג לפניכם את האפליקציה שפיתחנו SUPER QUICK למציאת מוצרים בחנות ולחישוב מסלול אופטימלי לאיסוף מוצרים. |
|  | נתבונן בחנות הזו. כדי למצוא במבה, שמפו וקולה יצטרך הלקוח להסתובב סיבובים רבים ועד שיגיע לקופה יגמר לו הכוח.  הוא היה שמח אם משהו היה מתנדב לעזור לו ולהראות לו איך ללכת. |
|  | המטרה שלנו להפוך את הקניה ממעצבנת ומסובכת לחוויה כיפית, פשוטה ונעימה.  **1.** שימוש באפליקציה שלנו תחסוך זמן בשיטוט מיותר.  **2.** פיתחנו מוצר פרקטי המאפשר לכל בעל חנות למפות את הנתונים שלו ולהנגיש אותם ללקוחות [ללא דרישות מיוחדות כמו התקנת שבבים]  **3.** המוצר מתאים לווריאציות רבות, יכול להיות אפליקציה או אתר, וכן כעמדת מחשב בחנות. הוא יכול גם להשתלב במכשירים הקיימים כיום בחנויות כמו מסופונים.  **4.** הקפדנו על ממשק נח כך שהלקוח יכול בקלות לבחור מוצרים ולאתר אותם במפת החנות. |
|  | **האתגרים שעמדו לפנינו:**  **1.** מיפוי נתוני החנות, ז"א להחליט איזה נתונים נצטרך מבעל החנות כדי שנוכל לחשב מרחקים.  נתייחס אל החנות כגרף בעל קודקודים וקשתות וכך נוכל להשתמש באלגוריתמים של גרפים |
| K:\מה הכי חדש\מצגות\מצגת פתיחה ערוך פסי\שקופית7.JPG | במקום שכל מוצר יהיה יעד, כמה מוצרים סמוכים יהיו ליעד אחד. |
|  | **3.** היינו צריכות לחשב את המרחקים ולמלא מטריצת מרחקים, וזה היה שלב שדרש ריכוז והעמקה. |
|  | **2.** מימוש אלגוריתמים דייקסטרה וTSP שהם אלגוריתמים ידועים, אך היינו צריכות להפוך את הפסאו קוד והמלל לקוד שעובד ומחזיר תוצאות נכונות ומתאים לנתונים שלנו.  על TSP קראנו הרבה חומר, ומימשנו כמה מהרעיונות שמצאנו באופן עצמאי בהתאם לצרכים שלנו. |
|  | **4.** ולאחר שנדע מפונקציית הTSP מה סדר איסוף המוצרים היינו צריכות לדעת איך מגיעים ממוצר למוצר, איך לשרטט את המסלול. |
|  | 5. אתגר נוסף היה הצגת מפת החנות ללקוח.  רצינו לצייר ללקוח את החנות בצורה היפה ביותר, שתהיה ברורה ומסודרת.  חיפשנו זמן רב וחקרנו על אפשרויות שונות, ולבסוף בחרנו לצייר את החנות בעזרת קנווס. היינו צריכות ללמוד על האפשרויות שלו. לא מצאנו עליו הרבה מידע, קראנו את הפונקציות שמספק עד שהגענו לתוצאה שהתאימה לנו. |
|  | אז זה התהליך:  הלקוח בוחר מוצרים לקנייה, ננסה לקבץ מוצרים קרובים שיהיו נקודה אחת, נחשב את המרחקים ביניהם, נפעיל פונקציית TSP שתקבע את סדר איסוף המוצרים, ונצייר ללקוח את המסלול במפת החנות. |

אנחנו רוצות להראות לכם דוגמא.

|  |  |
| --- | --- |
| L:\גיבוי מחשב נייד\פסי תכנות\גרסאות פרויקט\21\מסלול עם התפצלויות.png | בחנות הזו בחרנו 13 מוצרים. המוצרים נמצאים ב5 סטנדים. אפשר לקבץ את 2 הסטנדים האלה לאזור אחד ואת 2 הסטנדים האלו לאזור, וביחד עם המים יש לנו 3 אזורים.  נוסיף את נקודת ההתחלה- מיקום הלקוח ונקודות הסיום- הקופות ונקבל 6 אזורים.  נרצה לדעת את המסלול, איזה אזור יהיה ראשון ומי שני וכו' בעזרת הTSP, שצריכה לקבל מרחקים בין האזורים. |
|  | ביצירת החנות חושבו המרחקים בין נקודות הגישה בחנות. |
|  | נחשב מרחקים בין האזורים באמצעות המרחקים בין נקודות הגישה. |
|  | נפעיל את אלגוריתם TSP למציאת מסלול [לא עוברים על הקוד, סתם אולי שניה אחת נראה שמפעילים אותו והאינדקסים שחזרו] ו- חזר לנו הסדר. |
| L:\גיבוי מחשב נייד\פסי תכנות\גרסאות פרויקט\21\מסלול עם התפצלויות.png | נצייר ללקוח מסלול.  איך הגענו מאזור לאזור בצורה יפה ולא מתחנו קו אחד מאזור לאזור?  במטריצת המרחקים בעבור כל אזור מצויין הדרך דרכה נגיע לכל אזור יעד, וכך ציירנו מסלול ברור ונאה למשתמש. |

עכשיו אנחנו פנויות לשאלות, או להראות עוד דברים באפליקציה המדהימה שלנו!!!

נשאל אותם מה מעניין אותם:

* איך חישבנו את המטריצת מרחקים
* חישוב מעבר בין האזורים
* פונקציות TSP
* פונקציית דייקסטרה