BGU

הפקולטה למדעי ההנדסה

המחלקה להנדסת תעשייה וניהול

אוטומציה וייצור ממוחשב 364-1-3321, סמסטר א' תשפ״א

**תרגיל מעבדה – Lab View – "My Dyson"**

(**שימו לב**: בסוף המסמך ולפני הנספחים מופיעות הערות חשובות והנחיות כלליות אנא קראו אותן בעיון והקפידו לבצע אותן, כמו כן יש לעיין בנספח 1 – דרישות לעבודה נכונה)

חברת Dyson"" השיקה שואב אבק רובוטי "My Dyson" עליו ניתן לשלוט מרחוק על מנת לנקות את הבית ללא קרבה לשאר בני המשפחה על מנת למנוע הידבקות מנגיף הקורונה. השואב משמש לניקוי הבית באופן אוטומטי ואף מגיע לאזורים שאחרים לא יכולים או מתקשים להגיע אליהם או לחדרים בהם שוהה אדם בבידוד. אתם נבחרתם לכתוב את תוכנת הניהול והבקרה של השואב!

באמצעות תוכנת הניהול שתכתבו, המשתמש יוכל לשלוט ולבקר את המערכות השונות של השואב: לנווט, להפעיל את ציוד הניקוי ולבצע פעולות נוספות.

ממשק המשתמש של השואב כולל מספר מסכים שונים (לאורך התרגיל מפורט הסבר המבהיר מהן הדרישות והציפיות עבור כל מסך):

|  |  |
| --- | --- |
| **שם המסך** | **מהות המסך** |
| ראשי (Main) | תפריט המכיל ניווט בין מסכים |
| אתחול (Initialize) | אתחול משתנים והרצה אוטומטית של השואב |
| ניקוי (Clean) | הפעלת השואב לניקוי השטח |

**מסך ה- Main**

מטרת המסך הינה לנווט בין המסכים השונים שבתוכנה, כאשר ניתן לעבוד בזמן נתון רק במסך אחד שנבחר, **שאר המסכים יהיו מנוטרלים** ולא יגיבו לשום שינוי. המידע שיוזן בכל אחד מהמסכים יועבר לשאר המסכים בעת יציאה מהמסך הנבחר.

**במסך זה יוצגו למשתמש הפרטים הבאים:**

1. **חיווי זמן ותאריך עדכניים** – על המסך הראשי יופיע חיווי בזמן אמת של התאריך והשעה העדכניים (מתעדכן כל שנייה).
2. **תמונות מתחלפות של השואב** – כל שניה תוצג תמונה שונה של השואב, 2 תמונות בשמות image1 ו- image2 (את התמונות ניתן למצוא בקבצים המצורפים לעבודה).

במסך זה יופיעו שלושה לחצנים :

* Initialize - פותח תת מסך הרלוונטי אליו
* Clean - פותח תת מסך הרלוונטי אליו
* Stop – עוצר את התכנית לגמרי (סיום שאיבה)

שימו לב, לא ניתן ללחוץ על מסך ה- Clean כל עוד לא ביצענו אתחול במסך ה- Initialize.

**מסך ה- Initialize**

במסך זה המשתמש מזין למערכת את הערכים השונים המאפיינים את עבודת השואב, כמו כן המשתמש יוכל להחליט האם ברצונו להפנות את הרובוט לנקודת העגינה (אשר נמצאת בראשית הצירים).

**במסך זה יוצגו למשתמש הפרטים הבאים:**

1. **פקד נומרי לבחירת כמות חדרים לניקוי** – בפקד זה המשתמש יזין את כמות החדרים שברצונו לנקות, ניתן לבחור לנקות לכל הפחות חדר אחד ולכל היותר 8 חדרים.
2. **פקד בוליאני לניקוי הסלון** – בפקד זה המשתמש יזין האם הוא מעוניין לנקות את הסלון ביחד עם החדרים.
3. **מיקום נוכחי של הרובוט בציר X ו- Y** : המתשמש מזין את המיקום הנוכחי של השואב בטווח ערכים שבין מינוס 200 לפלוס 200, במידה והוזן ערך הגבוהה מ 200 או קטן ממינוס 200, המחוון יציג את הערך הקרוב ביותר בטווח התקין (לדוגמה: המשתמש הזין מינוס 300, הערך שיופיע יהיה מינוס 200) .
4. **פקד Home** – פקד המאפשר למשתמש להנחות את השואב להתחיל לשאוב מנקודת העגינה - כלומר מיקומו בציר ה-X וציר ה-Y יהיה 0 בתחילת העבודה. פקד זה אמור לשמש כאשר הרובוט נמצא במיקום כלשהו שאינו ראשית הצירים ונרצה כי תחילת העבודה תהייה מראשית הצירים.
5. **RUN** – לחיצה של המשתמש על הפקד:  
   1. תנטרל את כל שאר הפקדים ותגרום לפקד ה- RUN להיעלם   
   2. תביא להופעה סרגל זמנים אשר יציג התקדמות של 5 שניות כאשר בתום הזמן תופיע על גבי המסך תוצאת החישוב של זמן העריכה של השואב לקראת ביצוע הניקוי (ראו תכנית לדוגמה), הזמן יחושב באופן הבא:

* כל חדר יהווה תוספת של 5 דקות לזמן הכולל
* זמן ניקוי הסלון הינו 15 דקות

זמן תנועה לראשית הצירים : סכימה של שני ערכי המיקום x,y (כל אחד מהם בערך מוחלט) כפול 0.2, בצורה הבאה: (|location x|+| location y|)\*0.2. זמן זה יחושב רק במידה ופקד HOME קיבל את הערך TRUE. הערכים שיכנסו לחישוב זה יהיה המיקומים שהוזנו בx,y. רק לאחר החישוב המיקום ישתנה ל0,0.

הזמן המשוער יוצג תחת הפקד **‘Total time’** (במספרים שלמים), פקד זה יהיה מוסתר מפני המשתמש בפתיחת המסך ויוצג למשתמש רק בתום שלב החישוב.

1. **EXIT** – יציאה לתוכנית הראשית. ערכי המיקום יועברו ממסך זה לשאר המסכים בעת היציאה ממסך זה, ויתעדכנו בעת כניסה מחודשת למסך על סמך הערכים שכבר היו מוזנים במערכת (כפי שמוצג בתוכנית לדוגמא).

**מסך ה- Clean**

במסך זה המשתמש מתפעל את תנועת השואב באופן ידני, כאשר כלל הנתונים שהוזנו עבור השואב ממסכים קודמים עוברים למסך זה.

**במסך זה של התוכנה יוצגו למשתמש הפקדים הבאים :**

1. כפתור Start.
2. כפתור Sleep.
3. כפתור חזרה למסך ראשי.
4. פקדים השולטים על כיוון תנועת השואב.
5. חיווי על מיקום השואב במרחב (בקואורדינטות של X,Y).
6. פקד ה- Flip.
7. פקד המציג את מצב העבודה בו נמצא השואב (mode).
8. מחוון המציג את מהירות הנסיעה.
9. חיווי על רמת כוח המשאבה של השואב.

**תיאור האלמנטים השונים במסך ה- Clean:**

1. **כפתור Start** – כפתור שלחיצה עליו מכינה את השואב למצב עבודה. על מנת שהשואב יהיה מסוגל לעבוד, עליו להביא את המשאבה למצב מוכנות מקסימלי של 100%. קצב התקדמות המוכנות הינו 10% לשנייה, בזמן זה לא ניתן לבצע שום פעולה אחרת עם השואב (כל שאר הפקדים יהיו מנוטרלים). כאשר השואב מגיעה ל-100% מוכנות יעלם כפתור ה-Start ובמקומו יופיע כפתור ה- **Sleep**.
2. כפתור **Sleep** (מצב שינה) – כפתור שלחיצה עליו תגרום לשואב לאבד את רמת מוכנות המשאבה לעבודה. הכפתור מופיע עם הגעת השואב לרמת מוכנות העבודה המקסימלית (100%). לחיצה על כפתור זה תגרום לרובוט להתחיל את התהליך ההפוך של ירידה ברמת המוכנות של המשאבה בקצב של 10% לשנייה עד ל- 0%, כאשר מגיע ל0% הכפתור יעלם. במצב מוכנות של 0% הכפתור יתחלף עם כפתור ה "Start" שיחזור להופיע. הכפתור יוצג רק כאשר השואב נמצא במצב מנוחה (ללא תנועה, mode=0, יוסבר בהמשך).
3. **כפתור חזרה למסך ראשי** – כפתור זה יחזיר את המשתמש למסך הראשי ויחד עימו יעביר את נתוני המיקום האחרונים של השואב. ניתן לחזור למסך הראשי כאשר השואב לא פועל ונמצא במצב של sleep או טרם תחילת העבודה, כל זמן שהרובוט נמצא בפעולה הכפתור יעלם.
4. **פקד המציג את מצב העבודה בו נמצא השואב (mode)** – לשואב ישנם 3 מצבי עבודה:

* מצב 0 – השואב נמצא במנוחה, מהירות 0, המשאבה אינה עובדת ומתמלאת בקצב של 2 יחידות כוח בשנייה, המילוי לא יתבצע בזמן שישנה התקלות במכשול (יוסבר בהמשך).
* מצב 1 – השואב במצב תנועה בלבד, כאשר מהירות התנועה הינה 5 יחידות והמשאבה אינה עובדת.
* מצב 2 – השואב במצב תנועה ושאיבה, מהירות התנועה הינה 3 יחידות והמשאבה עובדת וצורכת 5 יחידות כוח בשנייה.

כאשר השואב נמצא בתנועה (מצבים 1 או 2), הוא יתקל במכשול בסיכוי של 5%.   
על מנת להכריע האם השואב נתקל במכשול או לא, פעם בשנייה יוגרל מספר בין 0 ל-100. במידה והערך שהוגרל הוא בין 95 ל-100 סימן שהשואב נתקל במכשול, על מנת לעקוף את המכשול יש ללחוץ על פקד ה- **Flip**.

1. פקד ה- **Flip** - במקרה שבו השואב נתקל במכשול יופיעו שתי נורות (אדומה וירוקה) המתריעות על המקרה, הנורות יהבהבו לסירוגין (כפי שמוצג בתוכנית לדוגמא). השואב מפסיק את עבודתו, עובר למצב מנוחה (mode=0) ומהירות 0, כלומר השואב אינו יכול לנוע וממתין עד שיזיזו אותו (גורם לסיבוב של הרובוט) באופן ידני באמצעות לחיצה על פקד ה-Flip. במצב זה כל הפקדים מנוטרלים ובכדי לאפשר חזרה לעבודה יש צורך ללחוץ על פקד ה- **Flip** אשר ישיב את השואב **למצב העבודה האחרון** שהיה טרם התקלות במכשול. לאחר לחיצה על הפקד, נורות ההתרעה ייעלמו וניתן יהיה לשלוט שוב על מצב השואב ועל כיוון תנועתו במרחב.
2. **פקדים השולטים על כיוון תנועת השואב Joystick)) –** המשתמש יכול לשלוט על תנועת השואב במרחב בעזרת שלט רחוק, ניתן לנוע ב- 8 כיוונים שונים, על פי שושנת הרוחות. פקד זה הינו מסוג radio buttons (מידע נוסף לגבי הפקד ניתן למצוא דרך Help של התוכנה).
3. **חיווי על מיקום השואב במרחב (בקואורדינטות של X,Y)** – אלו שני מחוונים המציגים את המיקום של השואב במרחב בקואורדינטות X ו- Y. בתחילת התוכנית השואב נמצא במיקום שנקבע במסך ה- Initialize, כאשר השואב מתחיל לנוע המיקום מתעדכן על פי הכיוון הנבחר מתוך שושנת הרוחות ומהירות הנסיעה.   
   חישוב התקדמות התנועה של השואב לכל ציר יהיה באופן הבא:

תנועה בכיוונים הראשיים (צפון, דרום, מזרח, מערב) תקדם את השואב בקואורדינטה המתאימה ע"פ מהירות התנועה. תנועה באחד מהכיוונים המשניים (צפון-מזרח, דרום-מערב וכו') תקדם את השואב בשתי הקואורדינטות ע"פ החישוב הגיאומטרי הבא עבור ציר x,y:

\* שימו לב שהחישוב נעשה בראדיאנים.

דוגמה: בתנועה בכיוון צפון במהירות 1, בכל שניה קואורדינטה Y תגדל ב-1. לעומת זאת תנועה בכיוון צפון מזרח במהירות 1, כל שניה X יגדל ב-070.7 ו Y יגדל ב-070.7.

1. **מחוון המציג את מהירות הנסיעה -** מחוון המציג את מהירות תנועת השואב על פי מצב העבודה בו השואב נמצא. לחבר למסך הראשי
2. **חיווי על רמת כוח המשאבה של השואב** – זהו גרף שמציג את רמת כוח המשאבה של השואב בכל רגע, רמת הכח מושפעת מהפעילויות השונות של השואב כגון ניקיון, טעינה ופריקה.

ירידה ברמת הכח תמודל לפי העיקרון הבא:

תנועה ושאיבה במקביל צורכת 5 יחידות כוח בשנייה.

אם רמת מוכנות כוח השאיבה מגיעה למצב של 5 יח' ומטה השואב יעצור ויעבור למצב עבודה מנוטרל (0mode = ), שינוי הגרף יושהה למשך 3 שניות (ראו תכנית לדוגמה).

חיווי על רמת הכוח בזמן טעינה ופריקה הוסבר בכפתורים START ו- SLEEP.

פונקציה VISIBLE

**תכנית לדוגמא**

על מנת למנוע מקרים של אי הבנה לגבי הפונקציונליות הנדרשת מהתוכנית תקבלו ביחד עם מדריך זה תכנית עובדת (ללא הקוד כמובן).

התוכנית שתכתבו אמורה לעבוד בדיוק כמו התוכנית לדוגמא. בצורה זו תוכלו לשחק ולהתנסות עם התוכנית לדוגמא ולהבין בדיוק מה התוכנית שלכם אמורה לבצע (כל שמות המסכים והכפתורים במדריך זה מופיעים בעברית ואילו בתוכנית לדוגמא הם מופיעים באנגלית).

על מנת להריץ את התוכנית לדוגמא יש **להוריד את כל הקבצים לתיקיה מקומית** ולהריץ משם.

שימו לב כי אינכם נדרשים לעיצוב מחודש, על מסך התכנית של העבודה שלכם להראות כמו מסך התכנית להדגמה (אתם רשאים להחליף צבעים ולהשקיע בעיצוב נוסף, אך זוהי אינה דרישה).

**הערות חשובות**

* תזמונים בתוכנית – במידה והתבקשתם לבצע במהלך התוכנית משימות בתזמונים מוגדרים (פעם בשנייה וכו'), ניתן להסתמך על לולאות עם השהייה על מנת לבצע תזמונים אילו. ז"א להפעיל אירועים בלולאה עם השהייה (של שניה לדוגמא) ולהניח שהאירועים מתרחשים פעם בשנייה. (בפועל מכיוון שקיים זמן ביצוע ועיכובים בשל צרכי מחשוב ללולאה עלול לקחת יותר משניה על מנת להתבצע, לצורך התרגיל נתעלם מסטייה זו).
* הפעלת קבצים נלווים לתוכנית – במהלך התוכנית אתם נדרשים לעשות שימוש בקבצים חיצוניים כגון קובציaudio קובצי תמונה וקבצי אקסל. עליכם לקחת בחשבון בקביעת הנתיב (Path) לקובץ את האפשרות שהתוכנית שלכם תופעל על מחשב אחר משלכם ועל כן הנתיב לקבצים יהיה שונה מהנתיב שעל המחשב שלכם. עליכם לדאוג שהתוכנית שלכם תוכל לעבוד על **כל** מחשב, העיקר שקבציי העזר ימצאו באותה תיקייה כמו התוכנית שקוראת להם (התוכנית שלכם). ועל כן עליכם לשקול שימוש בנתיב יחסי ולא בנתיב מוחלט על מנת לקרוא לקבצים (בסוף המסמך הזה מופיע נספח "קריאה לקבצים באמצעות נתיב יחסי").

שימו לב, עבודה אשר לא תרוץ על המחשב של הבודק - תשלח אליכם לסידור מחדש ויורדו לכם נקודות.

* מקרי קיצון ובאגים בתוכנית לדוגמה – התוכנית לדוגמא כשמה כן היא לדוגמא בלבד והיא נועדה על מנת להבהיר בצורה הטובה ביותר את אשר אתם נדרשים לבצע. ועל כן **לא ניתנה** התייחסות לכל מקרי הקיצון וכמו כן יתכן והתוכנית מכילה באגים מסוימים, אין להסיק מכך שאם בתוכנית לדוגמא יש באג זאת אומרת שאתם יכולים לממש את הבאג הקיים בתוכנית שלכם. על התוכנית שלכם לפעול בצורה הטובה והחלקה ביותר תוך מתן התייחסות למקרי קיצון. בכל מקרה שקיים ספק בנוגע לבאג מסוים או למקרה קיצון מסוים אל תניחו הנחות אלא פתרו את הבאג או מקרה הקיצון ותנו התייחסות לכך בתיעוד הקוד. לחלופין פנו עם הבעיה לצוות הקורס על מנת לקבל תשובה ספציפית לבעיה (יינתן ניקוד על טיפול בבאגים ובמקרי קיצון)

**הנחיות כלליות:**

* קראו את הוראות העבודה לעומק על כלל פרקיה (כולל הנספחים).
* חובה להקפיד על חיווט מסודר, על מנת שיהיה ניתן להבין את קוד התוכנית.
* יש להתמודד עם כל מצבי הקיצון ואין להניח הנחות מקלות.
* הרצת התוכנית תבוצע על ידי לחיצה על כפתור RUN (חץ בודד).
* הרצת התוכנית תתבצע אך ורק מהפרוצדורה הראשית (Main) ולא ממסך אחר.
* יושם דגש מיוחד על הצגת מקרי הקיצון והטיפול בהם. כלומר, עליכם לתכנן ולתכנת את העבודה באופן בו תימנעו ממצבים שאינם נכונים לוגית.
* עליכם לכתוב קוד יעיל ככל הניתן.
* שני בני הזוג צריכים להיות מעורים בכל פרטי העבודה.
* **שמרו גיבויים וגרסאות של התוכנית באופן שוטף ב-drive שלכם. במידה והנכם עובדים באמצעות ה-vpn על מחשבים באוניברסיטה, שימו לב כי שמירת הפרויקט על המחשב הוירטואלי לא תישמר לכם לאחר שתתנתקו מה- vpn, הקפידו לגבות את הפרויקט בdrive האישי שלכם כל הזמן!**
* כל עדכון שיפורסם בפורום מחייב את כלל המגישים.
* התרגיל מכיל שימוש באלמנטים שלא נלמדו בתרגולים לשם בחינת יכולת התמודדות עצמית עם התוכנה.

בתוך לשונית קבצי עזר באתר הקורס יש תיקייה – "קבצי עזר LV" – 1.קובץ עזר לגרפים 2. דוגמא לתיעוד קוד, השתמשו בהם.

* **יש להקפיד על אתחול משתנים בכל אחד מחלקי התוכנית.**
* הקפידו לבדוק שהתוכנית רצה כמה פעמים ברציפות ללא שגיאות (חזרתיות).
* יש להקפיד על עיצוב בהצגת הנתונים (דיוקים אחרי הנקודה, אפסים מיותרים וכו').
* יש להקפיד שכל המסכים יופיעו במרכז המסך ובמידת הצורך במסך מלא (התוכנית שלכם צריכה להיראות ולהתנהג כמו התוכנית לדוגמא).
* יש להקפיד על עבודה נכונה כפי שמפורט בנספח 1 בפרק הנספחים.
* העבודה תיבדק על פי :
* הצלחת הריצה (מסכים, כפתורים, דמיון לתכנית לדוגמא, חזרתיות)
* קוד: מודולריות, שימוש בפונקציות
* התנהלות במקרי קצה

**הגשה:**

**בתיבת ההגשה במודל עד ה-12.11.20 בשעה 23:59.**

רק אחד מבני הזוג יעלה את העבודה, אין צורך להעלות פעמיים.

בנוסף, עליכם לצרף קובץ וורד עם תיעוד קוד (ראו הסבר בנספח 2)

**שימו לב – סטודנטים שלא יגישו את העבודה בדיוק לפי ההנחיות, יורדו להם נקודות.**

**בהצלחה!**

**נספחים**

**נספח 1 – דרישות לעבודה נכונה**

* לפני תחילת העבודה על התוכנה מומלץ להשקיע זמן בתכנון נכון כגון כתיבת תרשים זרימה ותכנון המבנה הכללי של התוכנית (פונקציות תתי תכניות וכו').
* נסו להבין לפני תחילת העבודה באילו אלמנטים (פונקציות) אתם עשויים להשתמש על מנת לבצע את דרישות התרגיל ולמדו איך להשתמש בכל פונקציה בנפרד, לדוגמא במידה ותראו שעל מנת לבצע את הדרישות אתם צריכים להשתמש בפונקציית חיפוש במערך, פיתחו תוכנית חדשה ב labview והציבו בה מערך עם כמה איברים ונסו להבין איך פונקציית החיפוש עובדת ללא רעשי רקע. לאחר שלמדתם איך להשתמש בפונקציה תוכלו לשלב אותה בתוכנית הכללית.
* קיימות ב labview כמה מטודות לקביעת התזמונים וסדר הפעולות בתוכנית (תזמון ע"י חיווט, sequence, state machine programing) נסו לבחור את השיטה הטובה ביותר בשבילכם אך קחו בחשבון שניתן לשלב בין השיטות השונות ובהחלט יכול להיות שנכתוב פונקציה אחת בשיטה אחת ואת האחרות בשיטה אחרת, קחו כל מקרה לגופו.
* עקרון הפעולה של פונקציות ב LabVIEW – הפונקציות ב-LabVIEW מתפקדות כמו בשפות תכנות אחרות ז"א שאם התוכנית קוראת לפונקציה מנקודה מסוימת בקוד הנקודה הזו נשמרת והתוכנית מבצעת את מה שמופיעה בפונקציה ומיד לאחר שהקוד של הפונקציה מסתיים התוכנה תחזור לאותה נקודה שממנה קראנו לפונקציה. **אין צורך לבצע פעולה מסוימת על מנת "לסגור" את הפונקציה אלא צריך לתכנת אותה כך שהיא תסתיים בצורה טבעית (כמו סיום של תכנית רגילה, אין להשתמש בפונקציות לסגירה כפויה של התכנית הקיימות בLabView).** במידה ולפונקציה שכתבנו יש ממשק עם המשתמש (הגדרנו שבעת הקריאה לפונקציה יפתח ה- Front panel ) והפונקציה שלנו סיימה את הריצה אך ממשק המשתמש לא נעלם, הפתרון לכך נמצא בהגדרות התצוגה של הפונקציה **ולא בכל מיני אמצעים (שקיימים אמנם) לסגירה כפויה.**
* פקדים בוליאניים – עבור כל משתנה בוליאני המוגדר כפקד (control) מוגדר אופן הפעולה המכני שלו. כאשר אופן הפעולה יכול להיות switch או letch. - switch הוא מצב שבו כל לחיצה על הפקד משנה את מצבו מ true ל false והוא נשאר באותו מצב עד ללחיצה הבאה, במצב letch לעומת זאת לחיצה על הפקד משנה את ערכו מ false ל true עד שהערך נקרא ע"י התוכנה ואז הוא חוזר אוטומטית למצב false. ניתן להגדיר בנוסף עבור כל מצב האם שינוי הערך יתבצע ברגע הלחיצה על הפקד או בזמן שחרור הלחיצה. כל פקד שנניח על מסך המשתמש יגיע עם הגדרת ברירת מחדל של אופן הפעולה המכני, במידה ונרצה לשנות מצב ניתן לשנות ע"י כניסה לתכונות (properties) של הפקד. עבור פקד שמצב הפעולה שלו מוגדר כ letch לא ניתן לייצר local variable.
* לכל משתנה ב LabVIEW יש רשימה של תכונות (properties), ניתן לשלוט על התכונות האלו במצב offline (לפני הרצת התוכנית) ע"י לחיצה על המשתנה במקש ימני ובתפריט לבחור properties. במידה ורוצים לשנות את אחת התכונות האלו בזמן הריצה ניתן להגדיר עבור כל משתנה אלמנט הנקרא property node וע"י אלמנט זה לשנות את התכונה הנבחרת בזמן ההרצה, על מנת לייצר property node יש לעמוד על המשתנה עם הסמן ללחוץ לחצן ימני -> create -> property node-> ומתוך הרשימה הנפתחת לבחור את התכונה הרצויה.
* על מנת להשיג עזרה בכל נושא שאתם לא מבינים ניתן להיעזר בעזרה של labview כמו כן ניתן למצוא פורומים שונים העוסקים בתוכנה ובהם ניתן למצוא הרבה תשובות.

<http://forums.ni.com>

<http://www.ni.com/getting-started/labview-basics/online-help>

<http://forums.ni.com/t5/LabVIEW/bd-p/170>

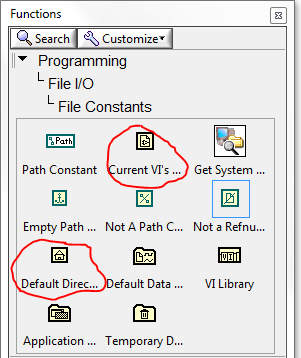
**נספח 2 - דרישות לתכולת הדו"ח.**

* שער- מכיל את תעודות זהות המגישים, שם העבודה, תאריך ההגשה.
* תוכן עניינים
* הקדמה ומטרת הפרויקט (עד חמש שורות): פיסקה קצרה המתארת את העבודה שנעשתה במסגרת התרגיל.
* הנחות יסוד (עד חצי עמוד): סעיף זה יכלול את כל ההנחות הנוספות שהנחתם (מעבר להגדרות התרגיל), עליהן מבוסס הפתרון שלכם (נא לא לחזור על הדרישות המובנות של תרגיל).
* תיאור מצבי הקיצון ושיטת הפתרון (עד שני עמודים): סעיף זה צריך לכלול הסבר לגבי כל המצבים והמקרים בתרגיל שהיה עליכם למצוא עבורם מענה פרטני שלא במסגרת האלגוריתם הכללי. אלו הם מצבים להם לא ניתן מענה בהגדרות התרגיל או בהנחות היסוד, והיה צורך למצוא עבורם פתרון ייחודי ולממשו בקוד התכנית.
* הגדרת משתנים: בתחילת כל מסך ראשי יש לרכז את כלל המשתנים בהם נעשה שימוש. הסבר קצר על כל משתנה, סוג וכיצד מאותחל.
* תיעוד קוד התכנית:
  + יש לצרף צילומי מסך של כלל הקוד בליווי הערות והסברים על התוכנית. על התיעוד להוסיף אינפורמציה ולהסביר את הקוד. אין לתרגם את הקוד כולו לעברית או לכתוב אותו ב"כתיבה מובנית" (Pseudo code) על צילומי המסך להיות ברורים וקריאים.

שימו לב שתיעוד הקוד צריך להסביר היטב את הקוד, אך עדין לשמור על תמצות.

* + נא לצרף צילומי מסך ברורים . השתמשו בפונקציית CROP או ב-Snipping tool.
* סיכום ומסקנות (עד חצי עמוד): פיסקה המתארת נקודות חשובות שנלמדו במהלך הפרויקט ומסכמת את העבודה שהתבצעה.

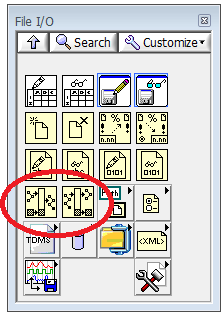
**נספח 3 – קריאה לקבצים באמצעות נתיב יחסי**

If you want to base your path according to a relative path, you can use a File constant such as Current VI’s path that it is inside the File IO and then wire it into the base input of the Build path.

Take in account that if you will generate an executable, the relative path is relative 

Please look at the following link for more explanation:

<http://digital.ni.com/public.nsf/allkb/FD7DE8BC8FFC256C862565F4006BE363>

****

**נספח 4 - שימוש במשתנה מסוג תפריט**

קיימים ב LabVIEW משתנים שבהם במקום שהמשתמש יקבל שדה ריק ובו יכניס את הערך המבוקש, מקבל המשתמש רשימה של ערכים והוא נדרש לבחור ערך מתוך הרשימה.

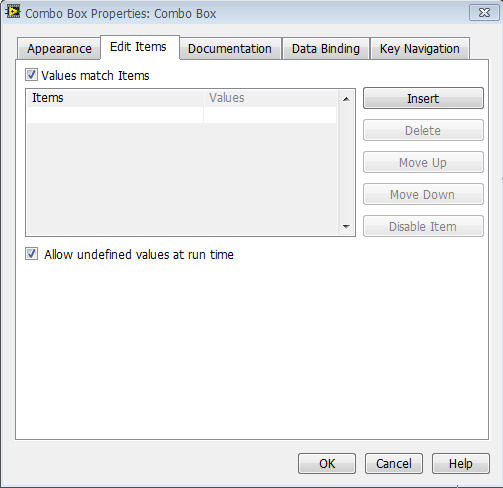
דוגמא למשתנה שכזה הוא ה Combo Box תמונה שמכילה מסך, מקורה, מרוצף, מטבח

התיאור נוצר באופן אוטומטי

על מנת להכניס את רשימת הערכים לתפריט נלחץ לחצן ימני על הפקד ונבחר מתוך התפריט שנפתח באפשרות “edit items”.

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

ובחלונית שתפתח נמלא את הערכים השונים באמצאות לחיצה על "insert". 

במידה ואנו מעוניינים למלא את רשימת הערכים און ליין בזמן הריצה של התוכנית נשתמש ב Propery Node.

ישנם משתני תפריט נוספים מסוגים שונים והם נמצאים בלוח הפקדים תחת התיקייה "Ring & Enum".